RNET ERP
(Material Request, Safety Module)

By

ADARSH PAWAR
[22MCA046]
VIVEK SANTANI
[22MCA053]

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
Ahmedabad 382481

RNET ERP
(Material Request, Safety Module)

Internship Report

Submitted in partial fulfillment of the requirements

For the degree of

Master of Computer Application

By

ADARSH PAWAR [22MCA046]
VIVEK SANTANI [22MCA053]

Guided By
Prof. Rajan Datt
[DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING]

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Ahmedabad 382481

CERTIFICATE

This is to certify that the Internship project entitled "RNET ERP (Material Request, Safety Module)" submitted by Adarsh Pawar 22MCA046 and Vivek Santani 22MCA053, towards the partial fulfillment of the requirements for the degree of Master of Computer Application of Nirma University is the record of work carried out by him/her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination.

Prof. Rajan Datt
Assistant Professor, MCA Department,
Computer Science and Engineering Dept.,
Institute of Technology,
Nirma University,
Ahmedabad

Dr. Madhuri Bhavsar,
Professor and HOD,
Computer Science and Engineering Dept.,
Institute of Technology,
Nirma University,
Ahmedabad

Riviera Infraprojects

Date: 25/05/2024

TO WHOM SO EVER IT MAY CONCERN

This is to certify that **Adarsh Pawar** (Roll No.22MCA046) of MCA Sem-IV, Institute of Technology, Nirma University, Ahmedabad has completed his Project Work from 1st January 2024 to 15th May 2024 with Riviera Infraprojects Pvt. Ltd, Ahmedabad on **developing "RNET ERP (Material Request, Safety Module)"** under the guidance of Mr. Amit Shah.

His willingness to learn and readiness to accept the challenges is appreciable. We recognize his valuable contribution and acknowledge the successful completion of his internship.

Regards,

Riviera Infraprojects Pvt. Ltd

RIVIERA INFRAPROJECTS AHMEDABAD

Authorized Signatory

079-71118181, 26931677
info@rivierainfra.com
www.rivierainfra.com

4th Floor, Goyal House,
Opp.Karnavati Club, S. G. Highway,Ahmedabad - 380015.
CIN U45309GJ2020PTC113328

Riviera Infraprojects

Date: 25/05/2024

TO WHOM SO EVER IT MAY CONCERN

This is to certify that **Vivek Santani** (Roll No.22MCA053) of MCA Sem-IV, Institute of Technology, Nirma University, Ahmedabad has completed his Project Work from 1st January 2024 to 15th May 2024 with Riviera Infraprojects Pvt. Ltd, Ahmedabad on **developing "RNET ERP (Material Request, Safety Module)"** under the guidance of Mr. Amit Shah.

His willingness to learn and readiness to accept the challenges is appreciable. We recognize his valuable contribution and acknowledge the successful completion of his internship.

Regards,

Riviera Infraprojects Pvt. Ltd

RIVIERA INFRAPROJECTS PVT. LTD. AHMEDABAD

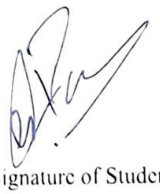
Authorized Signatory

079-71118181, 26931677
info@rivierainfra.com
www.rivierainfra.com

4th Floor, Goyal House,
Opp.Karnavati Club, S. G. Highway,Ahmedabad - 380015.
CIN U45309GJ2020PTC113328

Statement of Originality

I, Adarsh Pawar, Roll no: 22MCA046, give undertaking that the Internship Project entitled "RNET ERP(Material Request, Safety Module)" submitted by me, towards the partial fulfilment of the requirements for the degree of Master of Computer Application of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made. It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.



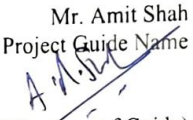
Signature of Student

Date: 24-05-2024

Place: Ahmedabad

Endorsed by

Mr. Amit Shah
Project Guide Name



(Signature of Guide)

Statement of Originality

I, Vivek Santani, Roll no: 22MCA053, give undertaking that the Internship Project entitled "RNET ERP(Material Request, Safety Module)" submitted by me, towards the partial fulfilment of the requirements for the degree of Master of Computer Application of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made. It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

Signature of Student

Date: 24-05-2024

Place: Ahmedabad

Endorsed by

Mr. Amit Shah
Project Guide Name

(Signature of Guide)

ACKNOWLEDGEMENT

My sincere thanks to Prof. Rajan Datt (Department of Computer Science and Engineering) for the unconditional and overwhelming support during the entire session of study and development, also, for guiding me throughout the internship period. All the faculties from our department provided us a favorable environment and necessary guidance, without them I would not have achieved our goal. They have always been available for us despite their busy schedule and were always a great source of inspiration for us.

I would like to extend heartiest thanks to Mr. Amit Shah(GM at Riviera Infraprojects Pvt. Ltd.), for supporting us during the internship period. He guided me all the time and motivated me within his busy schedule. I thank my senior colleagues Mr. Mrunal Jani and Mrs. Ripal Mevada for their guidance throughout the Internship period.

A blend of gratitude, pleasure and great satisfaction is what we feel to convey our indebtedness to all those who have directly and indirectly contributed to the successful completion of the project.

Thank you.

ABSTRACT/ Outline

The project aims to develop and enhance an in-house ERP system tailored to the specific needs of Riviera Infraprojects, a leading infrastructure development company. The ERP system serves as a comprehensive solution to streamline resource management, optimize operational efficiency, and facilitate informed decision-making across various departments and functions within the organization.

- **Background:**
    - Riviera Infraprojects currently relies on external ERP products for managing its resources, which poses challenges in terms of customization, integration, and scalability. Recognizing the need for a more tailored solution, the company has embarked on the development of an in-house ERP system to address its unique requirements and enhance overall operational effectiveness.
- **Scope:**
    - The scope of the project encompasses the entire lifecycle of designing, developing, implementing, and maintaining the ERP system for Riviera Infraprojects. Key components of the scope include requirements analysis, system design, development, integration, testing, deployment, and ongoing support.
- **Key Features and Enhancements:**

The proposed ERP system integrates various modules and functionalities, including but not limited to:

- **Admin:** User management, role-based access control, and system configuration.
- **Procurement:** Streamlined procurement processes from requisition to goods receipt.
- **Store:** Inventory management, stock control, and inventory valuation.
- **Sub-Contracts:** Management of subcontracting agreements, progress tracking.
- **Client-Billing:** Automated billing processes, invoice generation, milestone billing.
- **Finance:** Financial transactions, accounting processes, financial reporting.
- **Budget:** Budget planning, allocation, monitoring, and variance analysis.
- **Plant-Machinery**: Asset management, maintenance scheduling, depreciation calculation.
- **Safety and Quality:** Incident reporting, safety inspections, quality audits, compliance tracking.

Future enhancements include integration with BI tools for advanced reporting and analytics, the development of a mobile application for on-the-go access, and continuous improvement through user feedback and iterative development.

- **Expected Outcomes:**

Upon completion, the ERP system is expected to:

- Streamline resource management and optimize operational efficiency.
- Reduce dependency on external ERP products and improve customization.
- Standardize processes, enhance decision-making, and ensure compliance.
- Improve collaboration, communication, and transparency across the organization.
- Provide a scalable and flexible platform for future growth and innovation.

List of Figures

List of Tables

CONTENTS

Chapter 1: Introduction

1.1. About the company

1.1.1. Introduction of the company

- Riviera Infraprojects Private Limited (RIPL) stands at the forefront of the contracting industry, specializing in delivering comprehensive EPC projects, Core and Shell work, MEPF, and a variety of financing services.

- As an integral part of Goyal and Company, which boasts over five decades of distinguished experience in crafting landmark structures across Ahmedabad, Mumbai, and Bangalore, RIPL embodies a legacy of architectural excellence.

- Established in 2020, RIPL has quickly cemented its reputation in Gujarat and Karnataka, known for its robust expertise in the contracting field.

- The company offers seamless integration of services from the design phase right through to delivery.

- This broad spectrum of capabilities enables us to cater to diverse segments, including residential, commercial, industrial, and institutional projects. Join us in our mission to build not just structures, but a legacy of excellence that shapes the future of our cities.

- For more details: https://rivierainfra.com/

1.1.2. Quality Policy

At Riviera Infraprojects Pvt. Ltd., we are dedicated to consistent performance in terms of quality of services and the end product we deliver to achieve customer satisfaction. We strive for excellence through continuous improvement in our business processes.

As an organization, we are committed to:

- Complete the projects within the assigned timeline with essential quality parameters
- Continuous improvement in service quality through process control and best-in-class QMS to meet the expectations of customers
- Deploy state-of-the-art technologies and skilled employees
- Onboarding excellent Human Resources and their continuous training through reverse engineering
- Participate collaboratively for the development of our suppliers and contractors to enhance their capabilities of providing quality services
- Understand the voice of the customers leading us to Critical To Quality (CTQ) characteristics
- Managing costs through the elimination of inefficiencies in all business processes to prevent wastage

1.1.3. Communication

The company encourages us to communicate with our managers, colleagues, and subordinates in a way that is productive and contributes to the achievement of our company's goals. Meetings are a formal means of communication that is constant in all departments. Those who complete tasks that take place in multiple locations, and those who work independently, should attend meetings regularly.

We also hold regular meetings between department heads and company managers to ensure smooth functions. The company also arranges a special get-together informal chat session named “Coffee with Leaders” to have a light talk with leaders of the company every week.

Email Contact

- Business :- enquiry@rivierainfra.com

Call Us

- +91 79-26931677

1.1.4. Resources

- As Riviera Infraprojects is a service-based contractor construction company, it does not possess any IT products or other resources as such.

1.2.THE SYSTEM

1.2.1. Definition of the system

The Riviera Infraprojects ERP is an in-house ERP product of Riviera Infraprojects for managing the company's resources and reducing dependencies on external ERP products currently in use.

1.2.2. Purpose and Objective

- The purpose is to efficiently develop multiple components within the existing under-development system.
- Collaboration with the assigned development team under the guidance of the project guide.
- The components that are in the scope of the project are the Material Request and Safety module.
- The major objective is to Streamline resource management, optimize operational efficiency, and reduce wastage through an in-house ERP tailored to Riviera Infraprojects' needs.
- Decrease reliance on external ERP products by integrating necessary functionalities within the system, ensuring consistency, and establishing standardized procedures.

1.2.3. About present system

- **Admin Module**:
    - The Admin module serves as the backbone of the system, providing user management, role-based access control, and configuration settings for various modules.
- **Procurement Module**:
    - This module facilitates the procurement process, including requisition creation, supplier management, purchase order generation, and goods receipt.

- **Store Module**:
    - The Store module manages inventory and stock control, including stock entry, stock transfer, stock adjustment, and inventory valuation.
- **Sub-Contracts Module**:
    - This module handles sub-contract management, allowing the creation and tracking of subcontracting agreements, subcontractor management, and progress monitoring.
- **Client-Billing Module**:
    - The Client-Billing module automates the billing process for client projects, including invoice generation, milestone billing, and tracking of payments.
- **Finance Module**:
    - The Finance module manages financial transactions, accounting processes, and financial reporting. It includes features such as accounts payable, accounts receivable, general ledger, and financial statement generation.
- **Budget Module**:
    - This module facilitates budget planning, allocation, and monitoring. It allows users to create budgets for different projects or departments, track expenses against budgets, and generate budget variance reports.
- **Plant-Machinery Module**:
    - The Plant-Machinery module tracks the usage, maintenance, and scheduling of plant and machinery assets. It includes features for asset management, maintenance scheduling, and depreciation calculation.
- **Safety and Quality Module**:
    - This module focuses on managing safety and quality-related processes and compliance. It includes features for incident reporting, safety inspections, quality audits, corrective actions, and compliance tracking.

1.2.4. Proposed system

The proposed system builds upon the foundation of the current development phase, aiming to further enhance functionality, usability, and efficiency across all modules. Key features and improvements include:

- **Enhanced User Interface**: The proposed system will feature a modern and intuitive user interface, designed to improve user experience and streamline navigation across modules.

- **Advanced Reporting and Analytics**: Implementation of advanced reporting and analytics capabilities will empower users with real-time insights into various aspects of resource management, financial performance, and project progress.

- **Integration with Third-party Tools**: Integration with external tools and platforms will be expanded to facilitate seamless data exchange and collaboration with stakeholders, suppliers, and clients.

- **Mobile Access and Responsiveness**: The proposed system will prioritize mobile accessibility, allowing users to access critical functionalities and data from any device, anywhere, at any time.

- **Enhanced Security Features**: Robust security measures will be implemented to safeguard sensitive data and protect against potential cyber threats, ensuring compliance with industry regulations and standards.

- **Scalability and Flexibility**: The architecture of the proposed system will be designed for scalability and flexibility, allowing for easy customization and adaptation to evolving business needs and growth.

- **Streamlined Workflow Automation**: Workflow automation capabilities will be enhanced to further streamline approval processes, reduce manual intervention, and improve overall operational efficiency.

- **Continuous Improvement and Support**: The proposed system will be supported by a dedicated team committed to ongoing maintenance, updates, and continuous improvement based on user feedback and emerging industry trends.

- **Training and User Adoption**: Comprehensive training programs and user support resources will be provided to ensure smooth transition and maximum user adoption of the proposed system.

1.3.PROJECT PROFILE

1.3.1. Project Title

- RNET ERP (Material Request and Safety Module)

1.3.2. Scope of the project

- Efficient development of multiple components within the existing under-development system.
- Implement enhancements and updates to the ERP system based on feedback and evolving business requirements.
- Ensure the ERP system remains aligned with Riviera Infraprojects' goals and objectives over time.
- Collaboration with the assigned development team under the guidance of the project guide.
- The components that are in the scope of the project are Material Request and Weighbridge Integration.

1.3.2. Project Team

- Adarsh Pawar
- Vivek Santani

1.3.3. Hardware and software environment in the company

- 1 PC
- **Technology Stack:**
  - Frontend: Next.js, Typescript
  - Backend: Asp.net Core 6, C#
  - Database: PostgreSQL
- **Software Tools:**
  - Just In Mind (Wireframe Design Tool)
  - Visual Studio Code (Text Editor for frontend).
  - Visual Studio 2022 (SDE for backend).
  - PgAdmin (For accessing the database).
  - Postman (For testing APIs).

Chapter 2: System Analysis

2.1. **Feasibility Study**

The feasibility study is a critical step in the system development life cycle, as it evaluates the proposed project's viability and identifies potential risks or challenges that need to be addressed before proceeding with the implementation. The feasibility study for the Riviera Infraprojects ERP system involves the following aspects:

2.1.1. **Operational Feasibility**

The operational feasibility assessment determines whether the proposed system can be effectively integrated into the existing business operations and workflows of Riviera Infraprojects. It evaluates the system's compatibility with the company's organizational structure, culture, and operational processes.

One of the key operational feasibility factors for the Riviera Infraprojects ERP system is user acceptance. The system's success heavily relies on its acceptance and adoption by the end-users, such as the Store Manager, Site Engineers, Approvers, and other personnel involved in resource management, procurement, and project execution. To ensure operational feasibility, the development team conducted interviews, questionnaires, and site visits to gather requirements directly from the end-users. This approach helps ensure that the system aligns with their needs and existing workflows, increasing the likelihood of user acceptance and successful adoption.

Another operational feasibility consideration is the system's ability to streamline resource management, optimize operational efficiency, and reduce wastage. The ERP system aims to achieve these objectives by providing a centralized platform for managing various aspects of the construction process, such as material requests, weighbridge integration, procurement, subcontracts, client billing, finance, budgeting, and plant machinery management. By integrating these functionalities into a single system, Riviera Infraprojects can improve coordination, reduce redundancies, and enhance overall operational efficiency.

Moreover, the system's operational feasibility is supported by the company's commitment to transitioning towards an in-house ERP solution. This decision reflects Riviera Infraprojects' recognition of the potential benefits of a tailored system that can better meet their specific needs and reduce reliance on external ERP products.

2.1.2. Technical Feasibility

The technical feasibility assessment evaluates whether the proposed system can be implemented using the available technology, resources, and expertise within the organization or through external support.

The Riviera Infraprojects ERP system is technically feasible due to the selection of widely adopted and robust technologies, such as NextJs, TypeScript, ASP.NET Core 6, C#, and PostgreSQL. These technologies are supported by large and active communities, ensuring access to extensive documentation, libraries, and expertise. Additionally, the development team has the necessary skills and experience to work with these technologies, minimizing the risk of technical challenges or bottlenecks during implementation.

The system's technical feasibility is further supported by the modular design approach. The ERP system is divided into several components or modules, such as Admin, Procurement, Store, Sub-Contracts, Client-Billing, Finance, Budget, Plant-Machinery, and others. This modular architecture allows for incremental development, testing, and deployment, reducing the overall complexity and risk associated with implementing a monolithic system.

Furthermore, the integration of a workflow feature into the system enhances its technical feasibility. The workflow feature is designed to manage approvals throughout the system based on different categories, such as General, Item-Wise, and Category-Wise. This functionality ensures that the system adheres to the company's existing approval processes and hierarchies, minimizing the need for extensive process re-engineering and increasing the likelihood of successful adoption.

2.1.3. Financial and Economical Feasibility

The financial and economic feasibility assessment evaluates the project's costs, benefits, and return on investment (ROI) to determine whether it is financially viable for the organization.

From a financial perspective, the development of an in-house ERP system for Riviera Infraprojects can be considered feasible. By investing in its tailored system, the company can potentially reduce its dependence on external ERP products, which often come with recurring licensing costs or subscription fees. This cost-saving aspect contributes to the project's financial viability in the long run.

Additionally, the system's ability to optimize operational efficiency and reduce wastage can lead to significant cost savings for the company. By streamlining processes, minimizing redundancies, and improving resource management, the ERP system can potentially reduce operational costs associated with inefficiencies, material waste, and other overhead expenses.

Furthermore, the modular approach to system development allows for a phased implementation, where the most critical components or modules can be prioritized and implemented first. This incremental approach can help distribute the investment over time, making the project more financially manageable and reducing the upfront capital expenditure.

From an economic perspective, the ERP system's potential to enhance Riviera Infraprojects' competitive advantage and market position should be considered. By improving operational efficiency, reducing waste, and providing a centralized platform for resource management, the company can potentially gain a competitive edge in the construction industry, which can lead to increased profitability and growth opportunities.

Moreover, the system's tailored design to meet Riviera Infraprojects' specific needs can result in long-term economic benefits by reducing the company's reliance on external ERP solutions that may not fully align with its unique requirements.

2.1.4. Handling Infeasible Projects

While the Riviera Infraprojects ERP system appears feasible based on the initial assessment, it is essential to continually monitor and evaluate the project's feasibility throughout the development lifecycle. If any aspect of the project is deemed infeasible during the implementation phase, the following measures can be taken:

- **Identify the Root Cause**: Conduct a thorough analysis to identify the specific factors contributing to the project's infeasibility. This could involve reviewing the requirements, assessing the technical challenges, re-evaluating the financial projections, or analyzing the operational constraints.

- **Explore Alternatives**: Based on the identified root cause, explore alternative solutions or approaches that could address the infeasibility issues. This could involve modifying the system's scope, adjusting the implementation timeline, exploring alternative technologies or architectures, or re-evaluating the resource allocation.

- **Prioritize Critical Components**: If the overall project is deemed infeasible, consider prioritizing the development and implementation of the most critical components or modules that are essential for Riviera Infraprojects' operations. This approach can help deliver value to the organization while deferring or excluding less critical components.

- **Seek External Expertise**: If the infeasibility issues are related to technical or operational challenges, consider seeking external expertise or consulting services to address the gaps or provide guidance on overcoming the obstacles.

- **Re-evaluate Project Viability**: In cases where the project remains infeasible despite exploring alternatives and seeking external support, it may be necessary to re-evaluate the project's viability and consider terminating or significantly scaling down the scope to avoid further resource allocation and potential losses.

Effective communication and stakeholder involvement are crucial throughout the process of handling infeasible projects. Regular updates and discussions with key stakeholders,

such as project sponsors, end-users, and decision-makers, can ensure that informed decisions are made regarding the project's future direction.

2.2. **REQUIREMENT ANALYSIS**

The requirement analysis phase is a critical step in the system development life cycle, as it involves gathering, documenting, and analyzing the requirements for the proposed system. In the case of the Riviera Infraprojects ERP system, the requirement analysis process aimed to understand the company's existing processes, pain points, and desired functionalities to ensure that the system meets their specific needs.

2.2.1. Fact-Finding Techniques

To gather comprehensive and accurate requirements, the development team employed a combination of fact-finding techniques, including interviews, questionnaires, record reviews, and observations. These techniques were carefully selected to capture input from various stakeholders and gain a holistic understanding of the company's operations and requirements.

2.2.1.1. Interviews

Interviews were conducted with key stakeholders, such as the Store Manager, Site Engineers, Approvers, and other relevant personnel involved in resource management, procurement, and project execution. These interviews were crucial in gathering first-hand insights into the current processes, pain points, and desired features for the new ERP system.

During the interviews, open-ended questions were asked to encourage in-depth discussions and capture detailed requirements. The interviews covered topics such as:

- Current workflows and processes related to material requests, weighbridge integration, procurement, subcontracts, client billing, finance, budgeting, and plant machinery management.
- Challenges or bottlenecks faced in the existing processes.
- Pain points or areas where inefficiencies or wastage occur.
- Specific functionalities or features desired in the new ERP system.
- Approval hierarchies and authorization levels for various processes.
- Integration requirements with external systems or third-party components.
- User interface preferences and usability considerations.

The interviews were conducted in a structured yet flexible manner, allowing for follow-up questions and clarifications as needed. The insights gathered from these interviews played a crucial role in shaping the system's requirements and ensuring that it addresses the end-users needs effectively.

2.2.1.2. Questionnaires

In addition to interviews, questionnaires were distributed to relevant personnel across different departments and levels within the organization. The questionnaires were designed to capture quantitative and qualitative data related to the current processes, challenges, and desired features for the new ERP system.

The questionnaires covered a wide range of topics, including:

- Frequency and volume of material requests, procurement activities, and other transactions.
- Approval turnaround times and bottlenecks in the existing processes.
- Satisfaction levels with the current systems or procedures.
- Prioritization of desired features or functionalities.
- Perceived benefits of an integrated ERP system.
- Willingness to adopt and learn a new system.

The questionnaires were distributed both electronically and in paper form to ensure maximum participation and accessibility. Clear instructions and explanations were provided to ensure that the respondents understood the questions and provided accurate and meaningful responses.

2.2.1.3. Record Review

To gain a comprehensive understanding of the company's existing processes and procedures, the development team conducted a thorough review of relevant records and documentation. This included:

- Standard Operating Procedures (SOPs) related to material management, procurement, finance, and other relevant processes.
- Existing process manuals or workflow diagrams.
- Historical data and records related to material requests, procurement activities, and project execution.
- Financial records and budgeting documents.
- Contractual agreements with subcontractors and clients.
- Regulatory compliance requirements and industry standards.

By reviewing these records, the development team could identify areas for improvement, understand the underlying business rules and constraints, and ensure that the new ERP system aligns with the company's existing policies and regulations.

2.2.1.4. Observations

To supplement the insights gathered from interviews, questionnaires, and record reviews, the development team conducted on-site observations at the AES Project Site. These observations allowed the team to witness first-hand the daily operations, workflows, and interactions between various stakeholders involved in the construction process.

During the observations, the team took note of the following aspects:

- Physical movement of materials and resources across different locations and departments.
- Interactions between site engineers, store managers, procurement personnel, and other stakeholders.
- Challenges or inefficiencies encountered during material handling, transportation, or storage.
- Adherence to safety protocols and quality control measures.
- Communication channels and information flow between different teams or departments.
- Use of existing tools, systems, or software for managing operations.

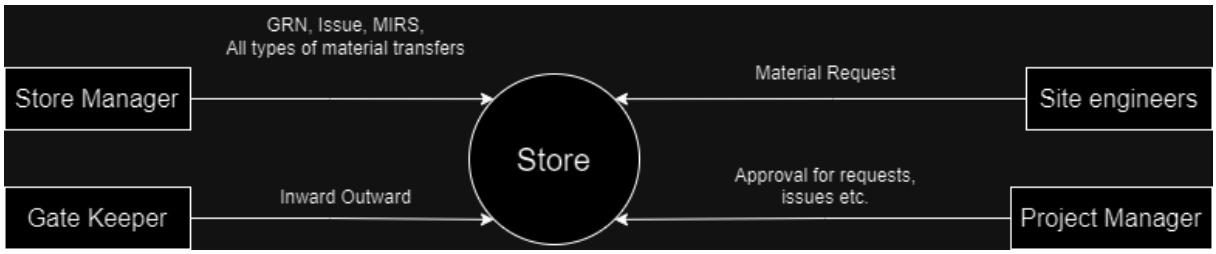
The observations provided valuable insights into the practical aspects of the company's operations, enabling the development team to identify potential areas for process optimization, automation, or integration within the new ERP system.

Throughout the requirement analysis phase, the development team maintained open communication channels with the stakeholders, encouraging feedback and clarifications. Regular meetings and discussions were held to validate the gathered requirements, ensure alignment with the company's objectives, and prioritize the implementation of critical functionalities.

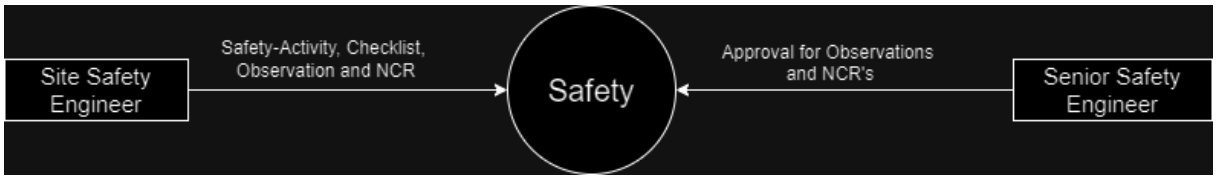
By employing a combination of fact-finding techniques, including interviews, questionnaires, record reviews, and observations, the development team was able to gather comprehensive and accurate requirements for the Riviera Infraprojects ERP system. This thorough approach laid a solid foundation for the subsequent phases of system design, development, and implementation, increasing the likelihood of delivering a system that meets the company's needs and contributes to improved operational efficiency and resource management.

2.3. **CONTEXT DIAGRAM**

2.3.1. Material Request Context Diagram



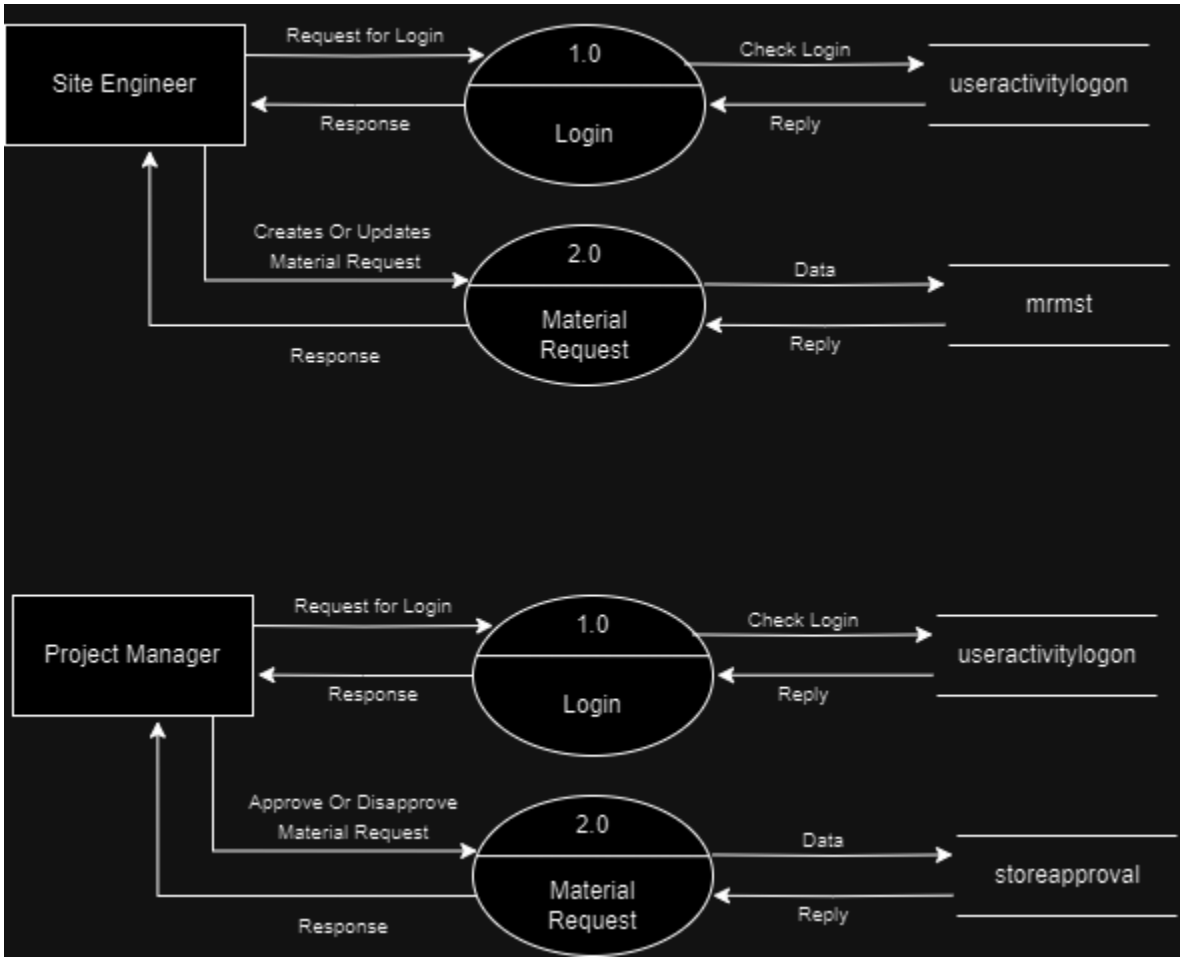
2.3.2. Safety Context Diagram



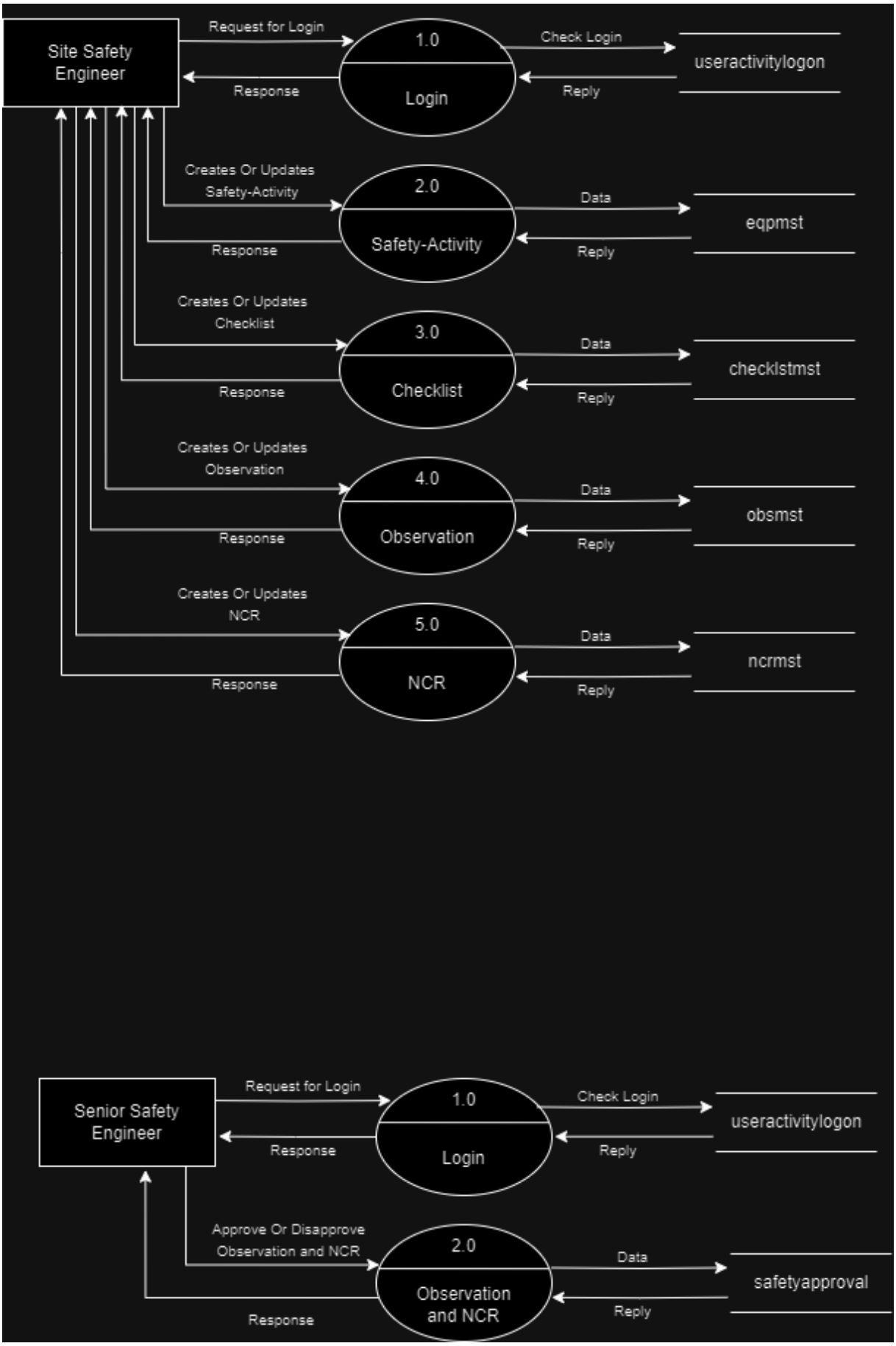
2.4. DATA FLOW DIAGRAMS

2.4.1. First level DFD

2.4.1.1. Material Request



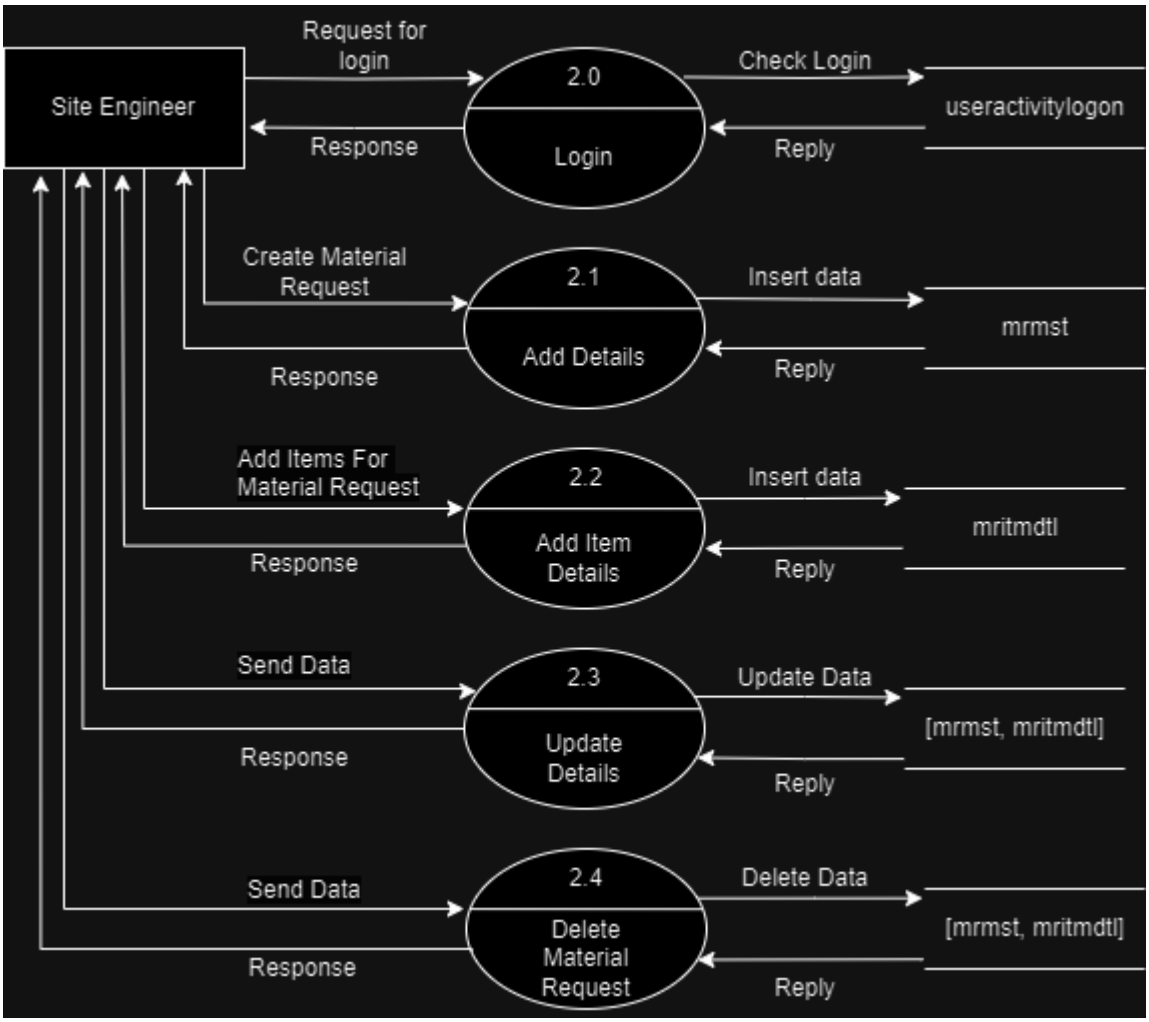
2.4.1.2. Safety


Site Safety Engineer
Request for Login
Response
1.0
Login
Check Login
Reply
useractivitylogon
Creates Or Updates Safety-Activity
Response
2.0
Safety-Activity
Data
Reply
eqpmst
Creates Or Updates Checklist
Response
3.0
Checklist
Data
Reply
checklstmst
Creates Or Updates Observation
Response
4.0
Observation
Data
Reply
obsmst
Creates Or Updates NCR
Response
5.0
NCR
Data
Reply
ncrmst
Senior Safety Engineer
Request for Login
Response
1.0
Login
Check Login
Reply
useractivitylogon
Approve Or Disapprove Observation and NCR
Response
2.0
Observation and NCR
Data
Reply
safetyapproval

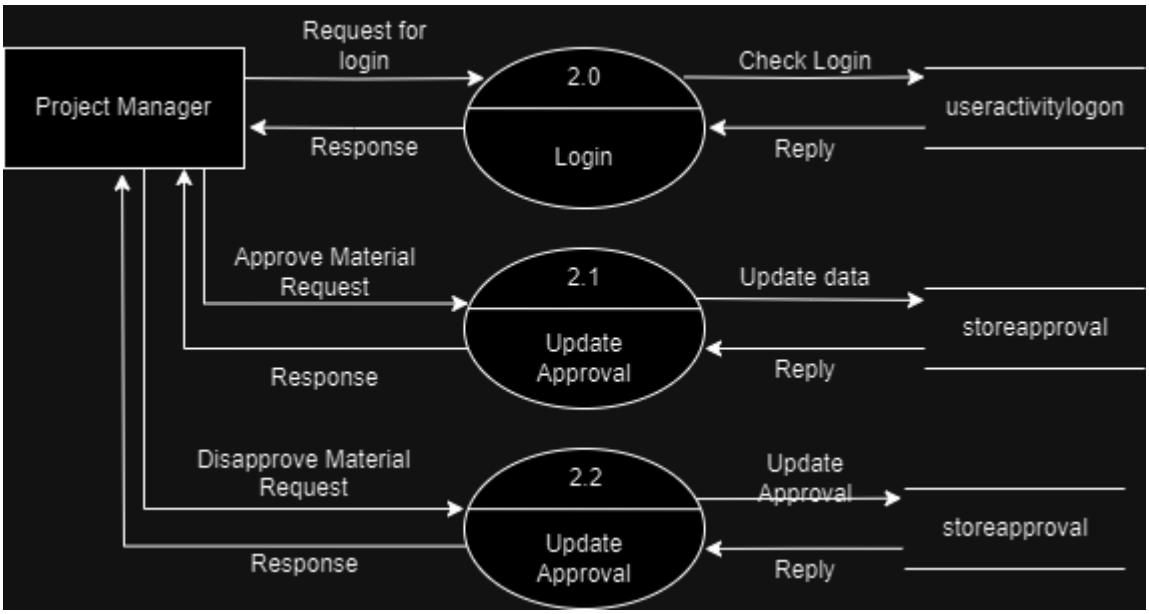
2.4.2. Second level DFD

2.4.2.1. Material Request

2.4.2.1.1. Site Engineer

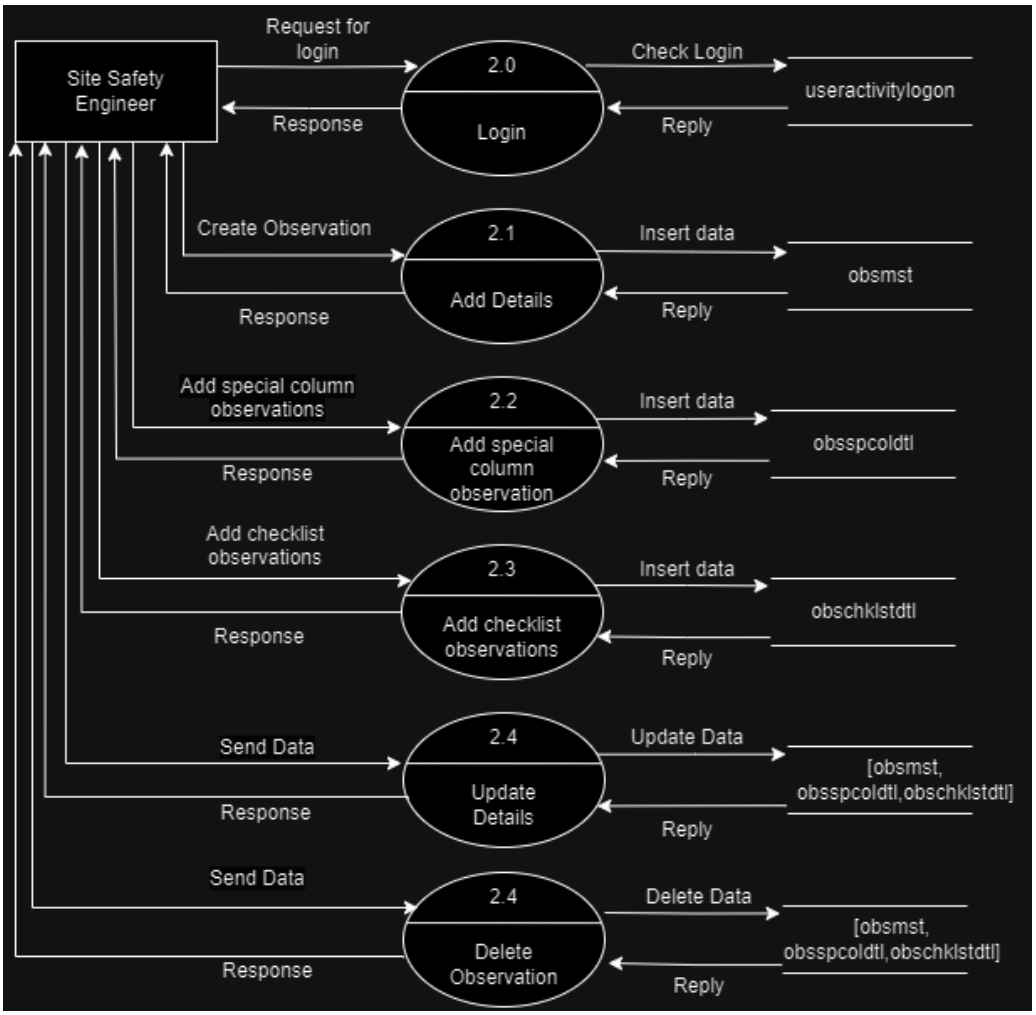


2.4.2.1.2. Project Manager

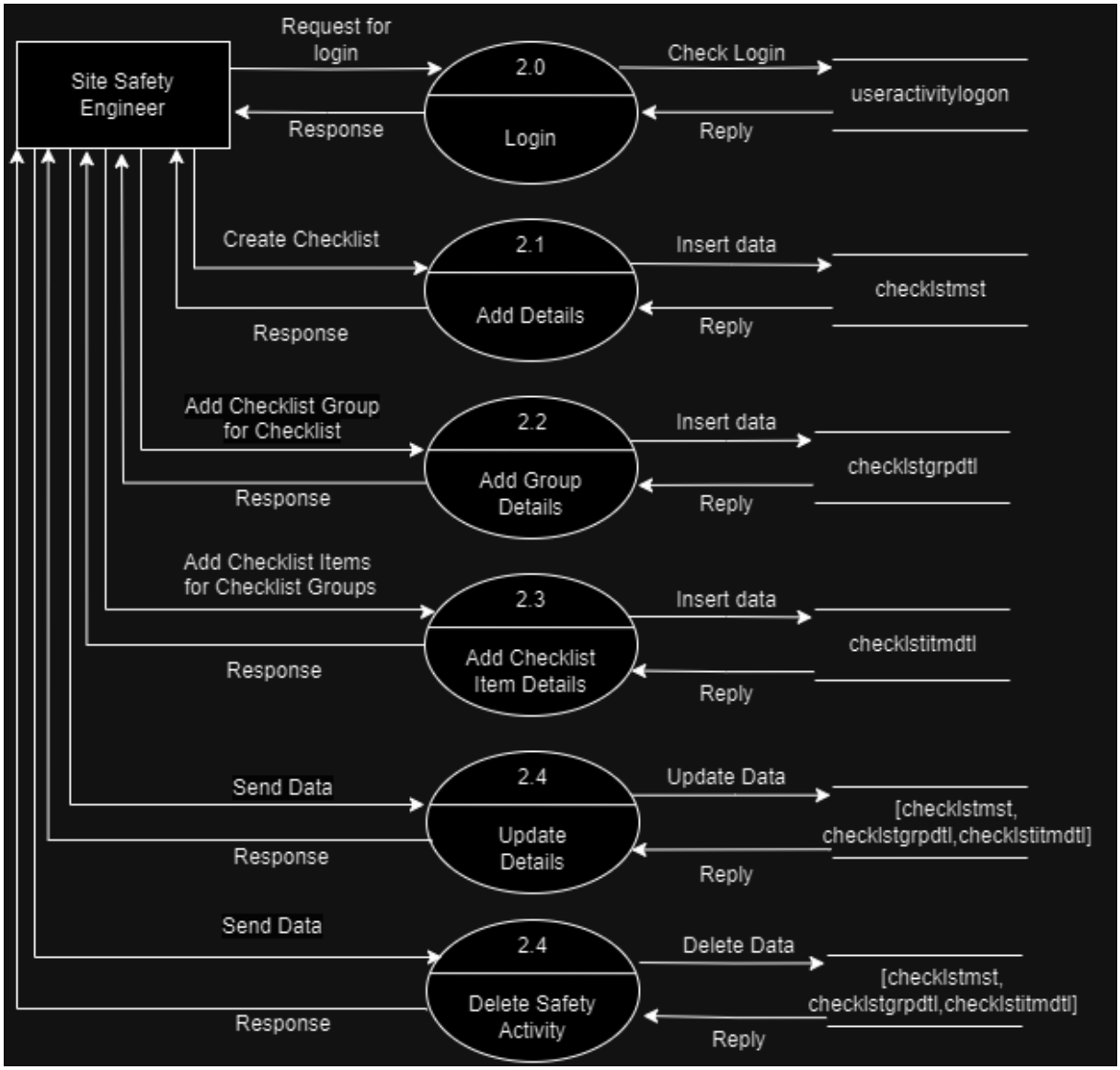


2.4.2.2. Safety

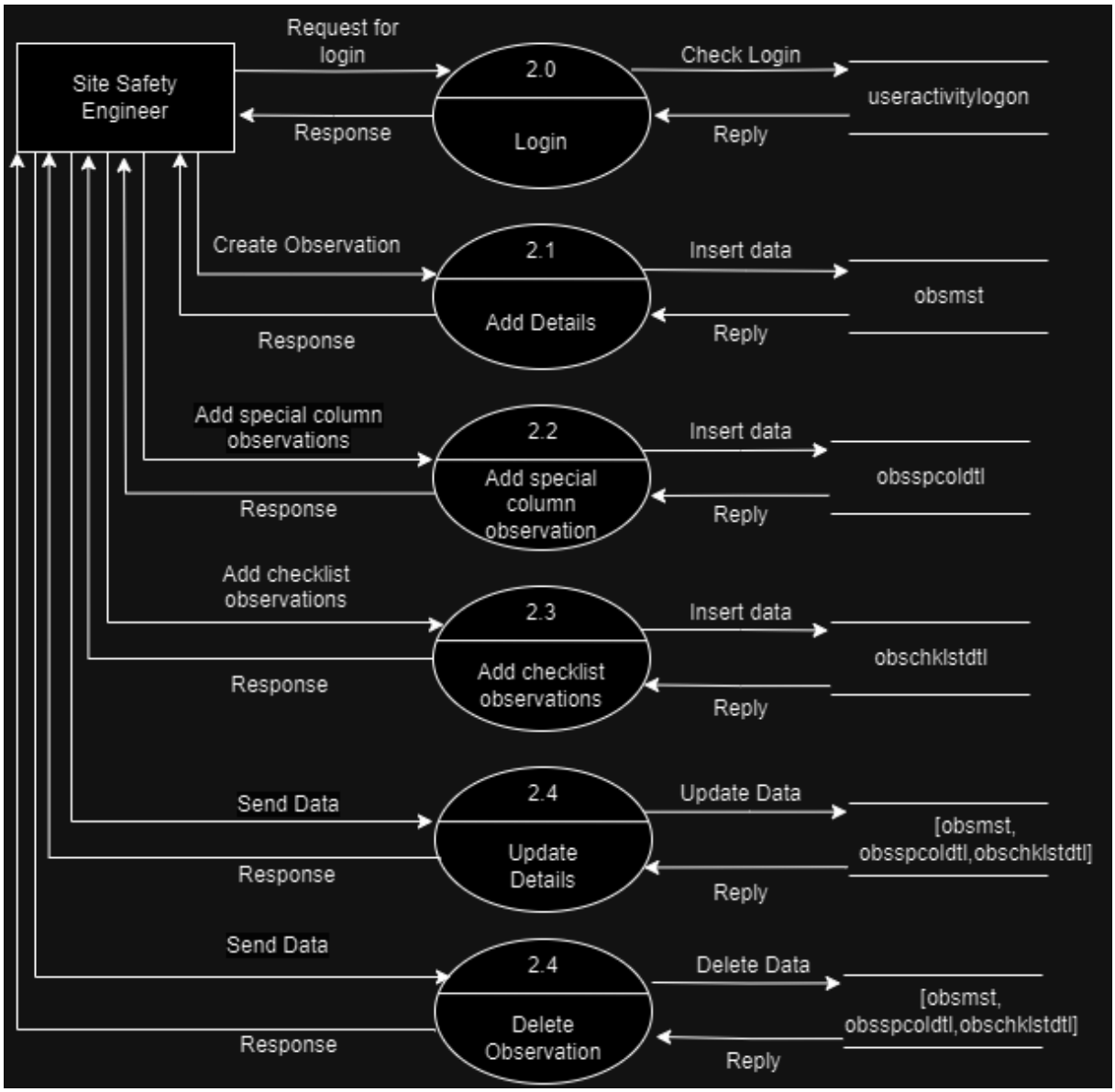
2.4.2.2.1. Site Safety Engineer Safety-Activity



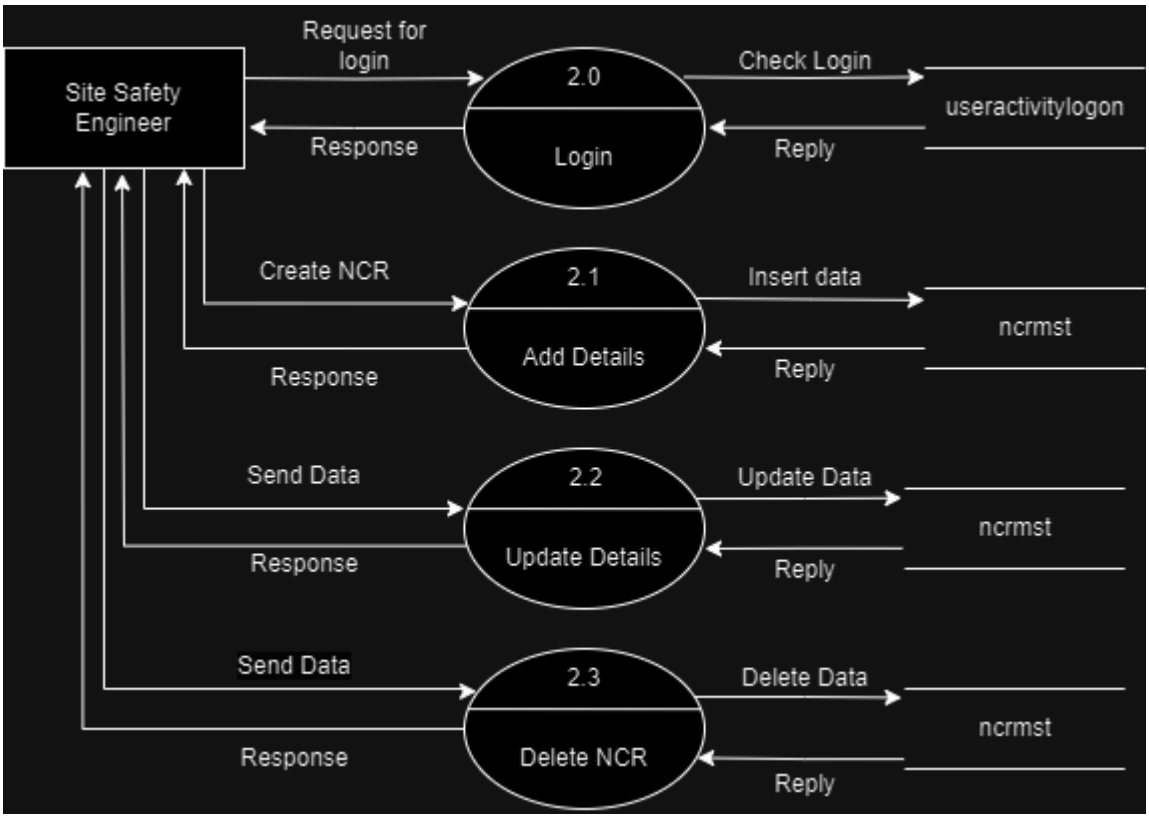
2.4.2.2.2. Site Safety Engineer Checklist



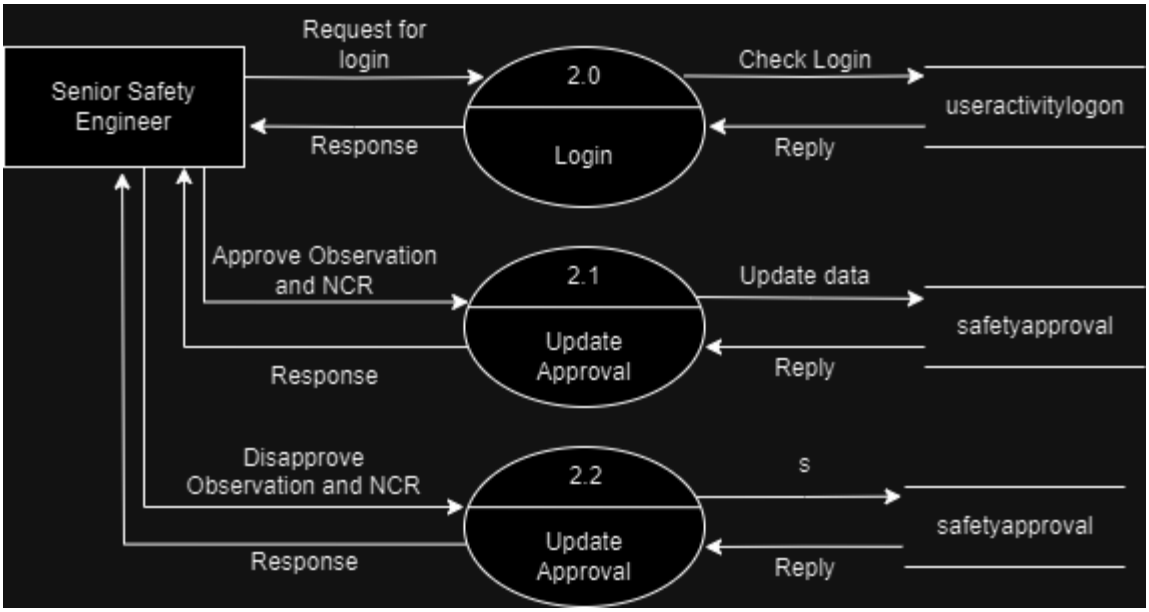
2.4.2.2.3. Site Safety Engineer Observation



2.4.2.2.4. Site Safety Engineer NCR



2.4.2.2.5. Senior Safety Engineer

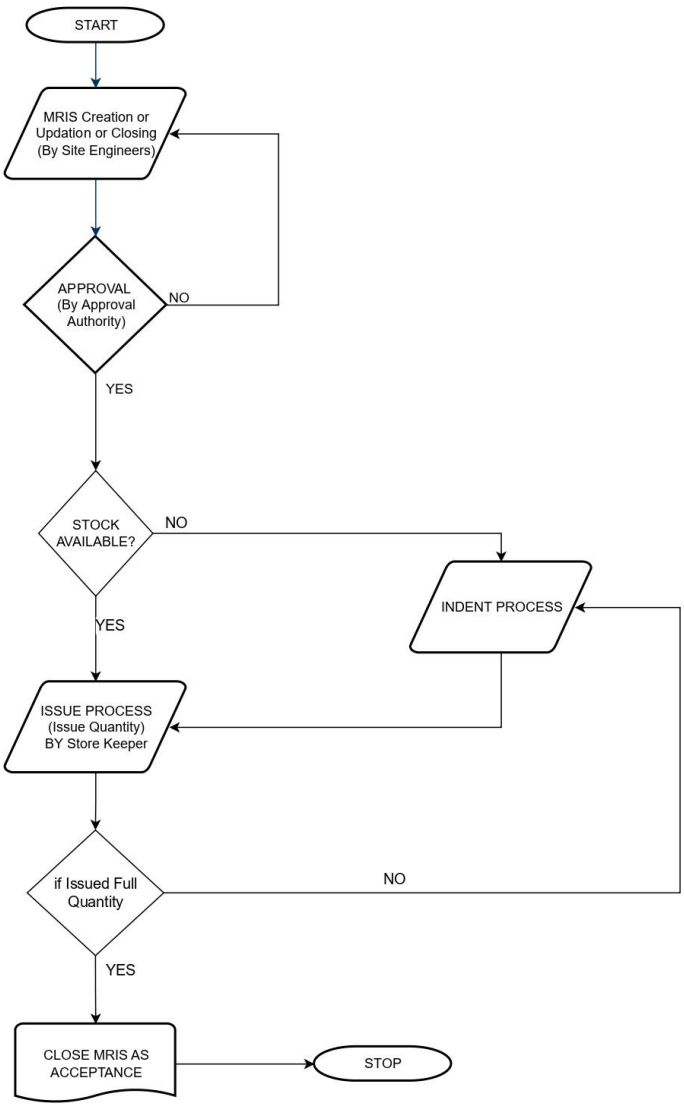


Chapter 3: System Design

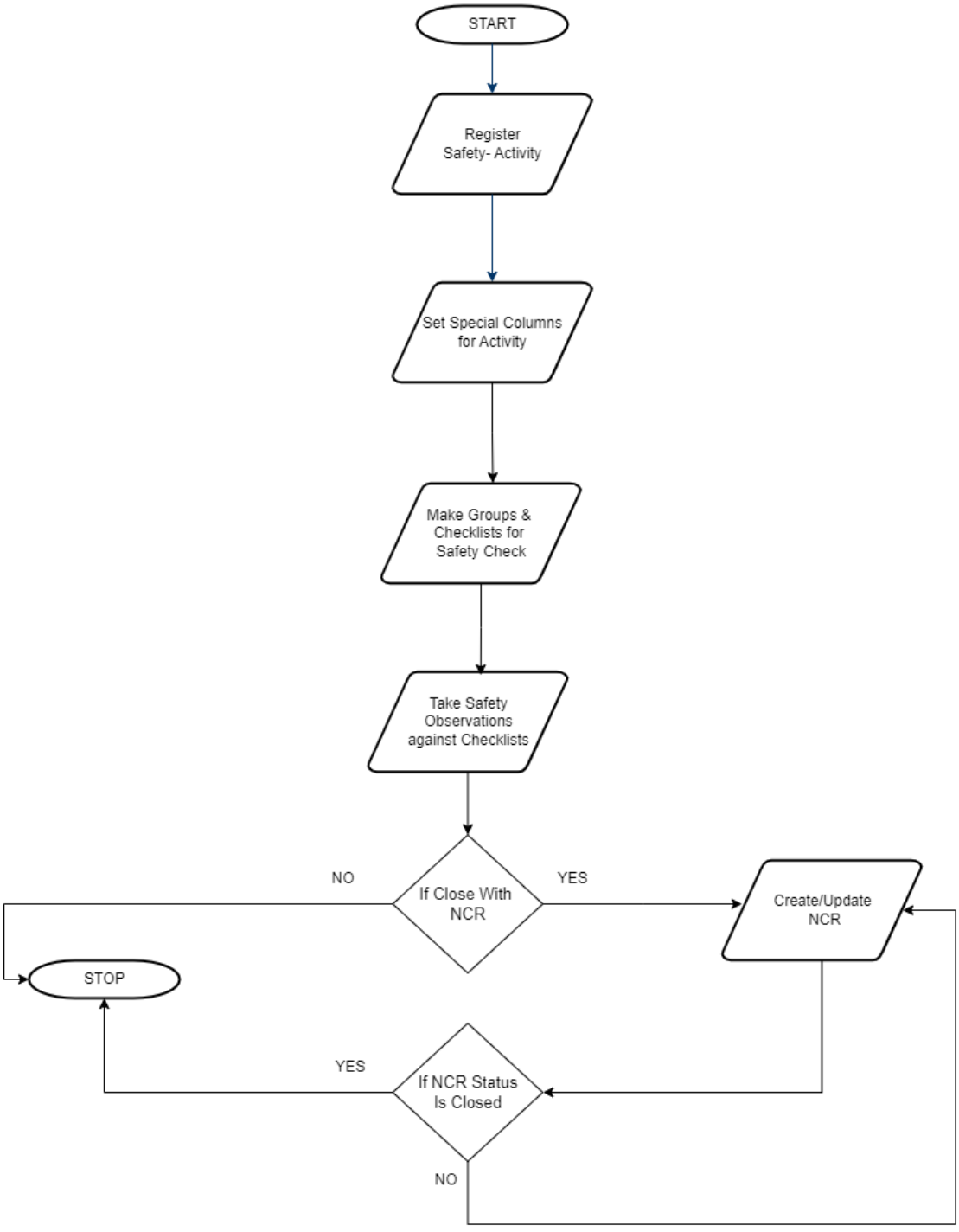
3.1. System Flow

3.1.1. Flowchart

3.1.1.1. Material Request

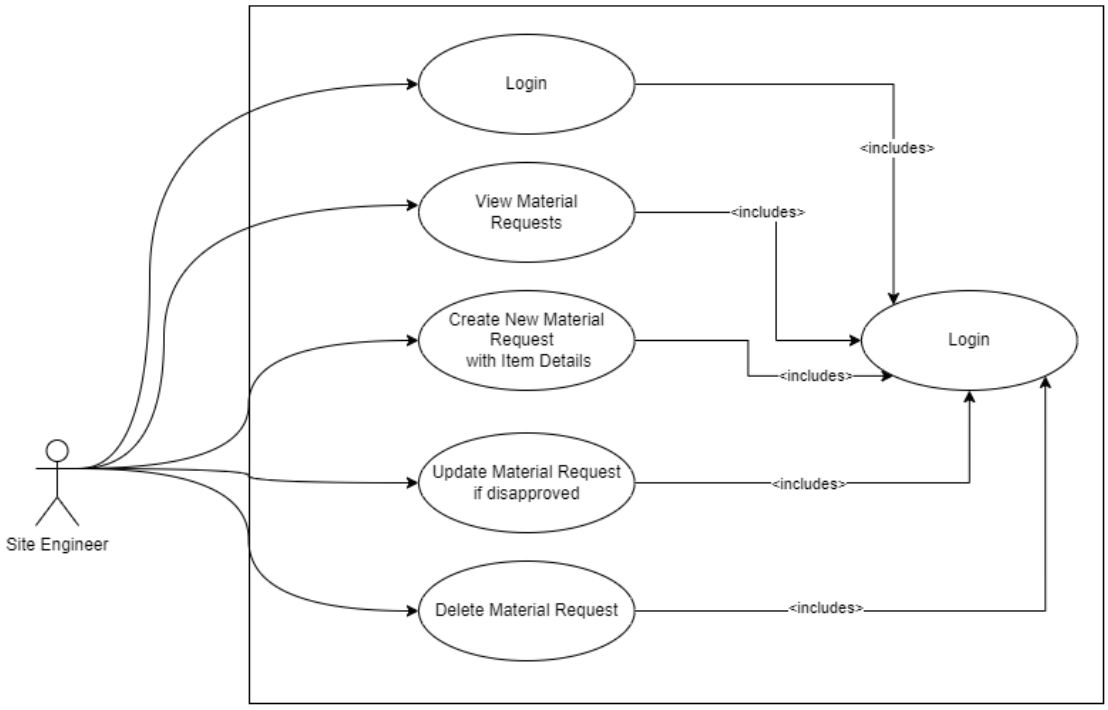


3.1.1.2. Safety Module

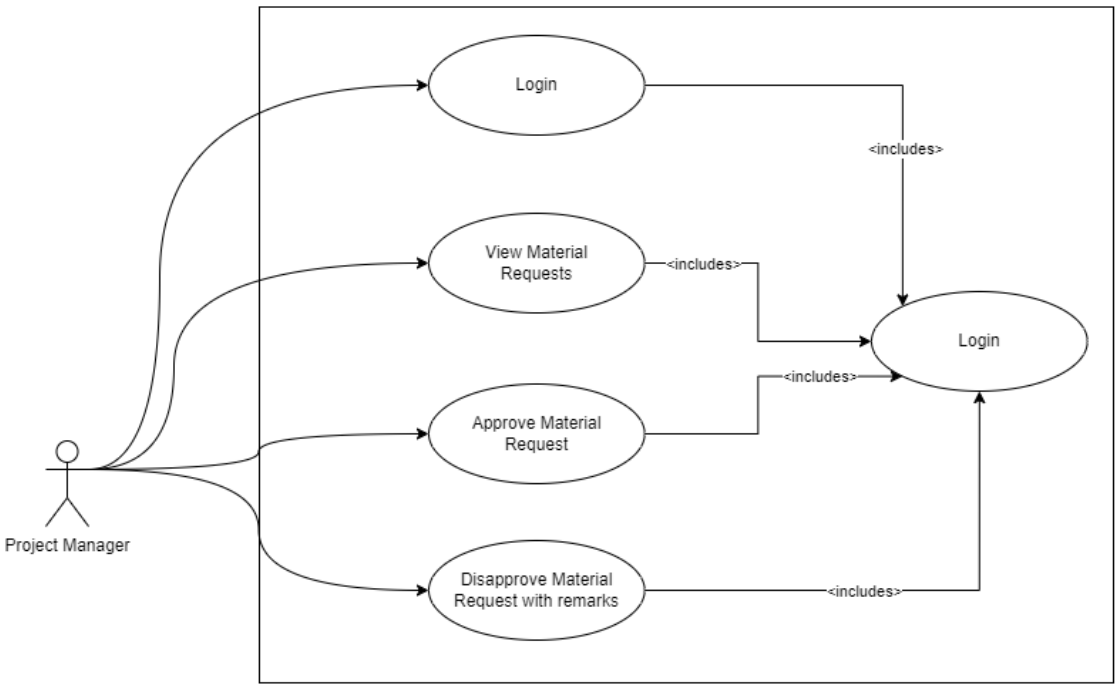


3.1.2. Use Case Diagram

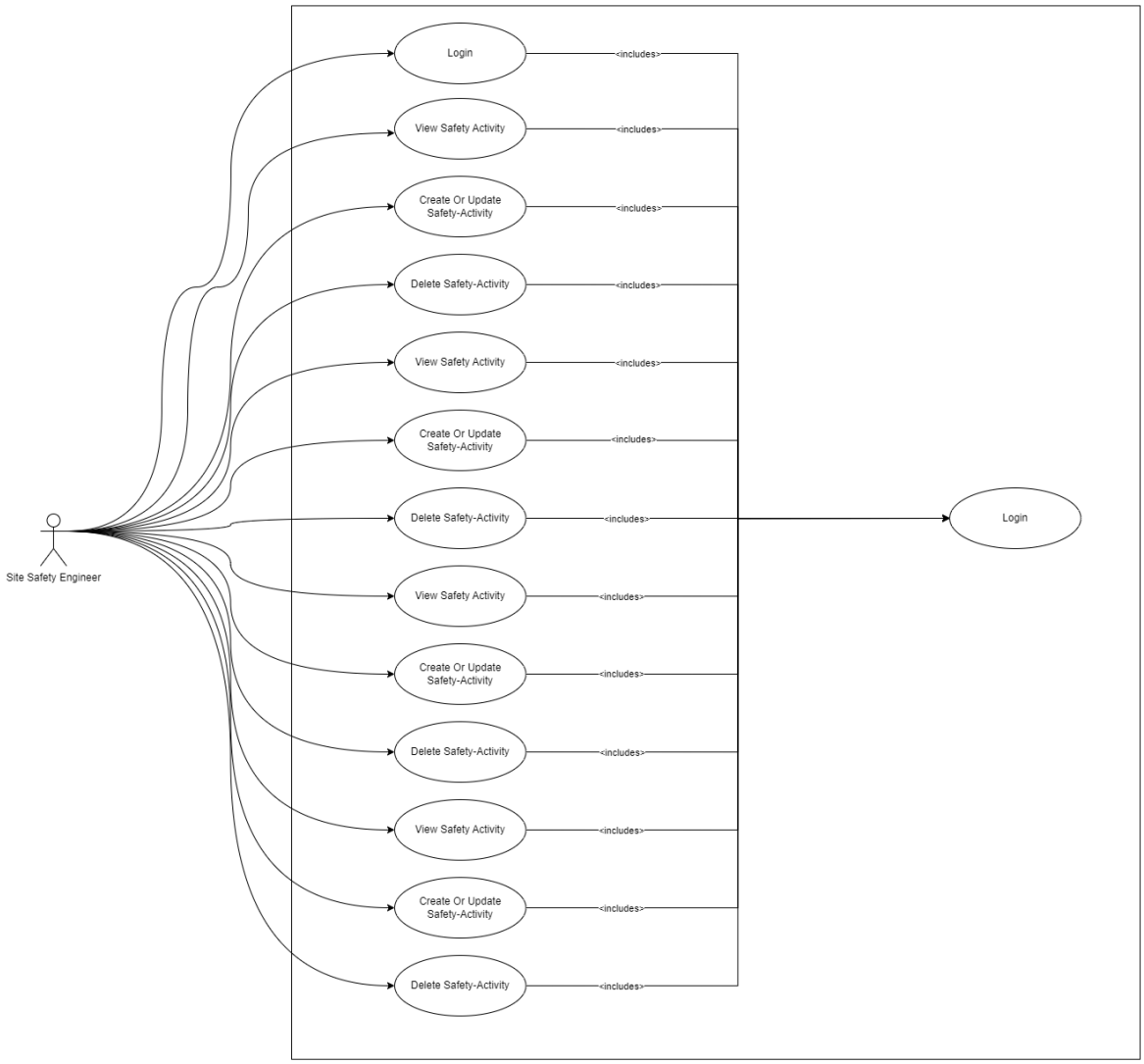
3.1.2.1. Store Site Engineer



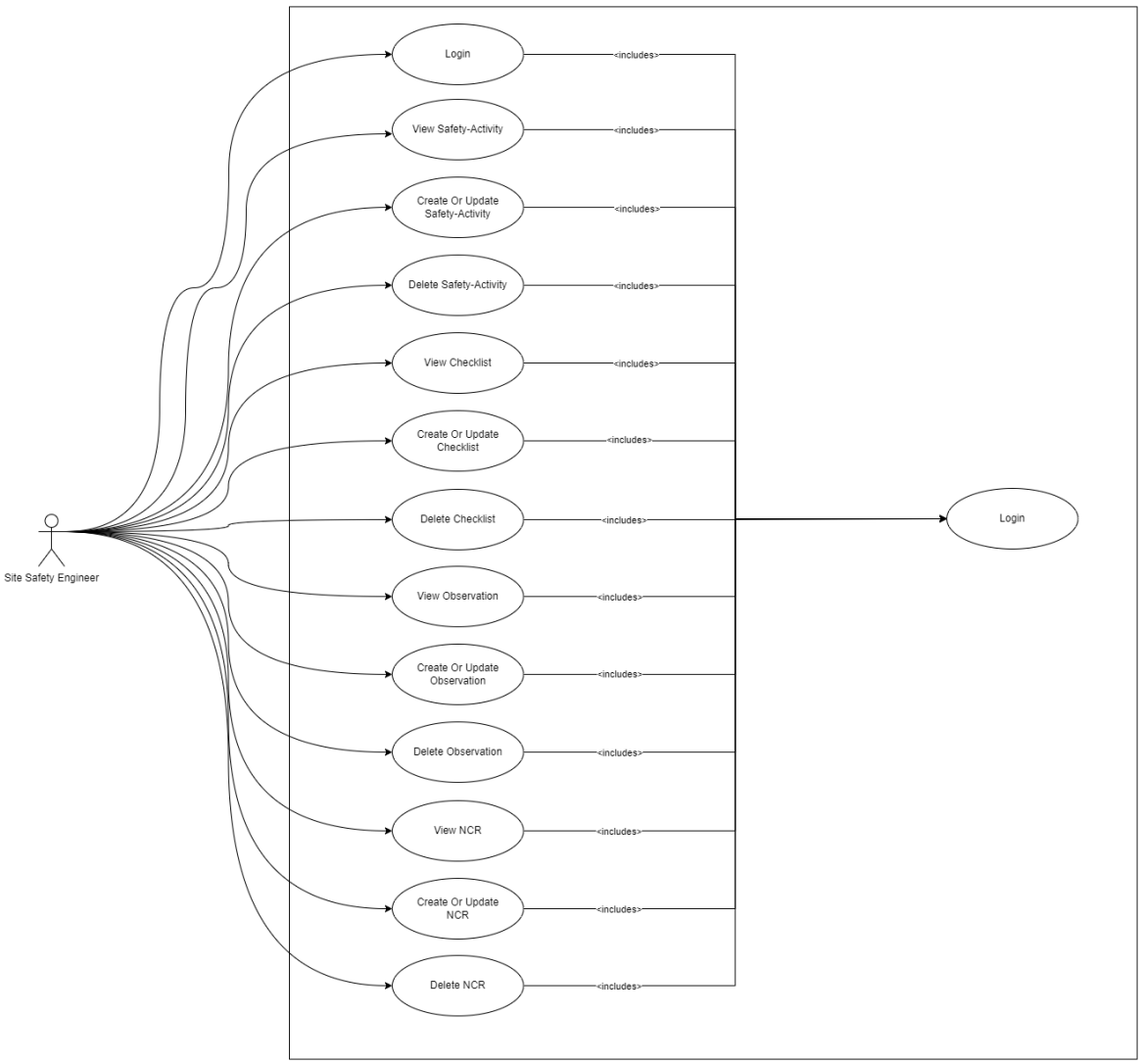
3.1.2.2. Store Project Manager



3.1.2.3. Site Safety Engineer (Safety Module)



3.1.2.4. Senior Safety Engineer (Safety Module)

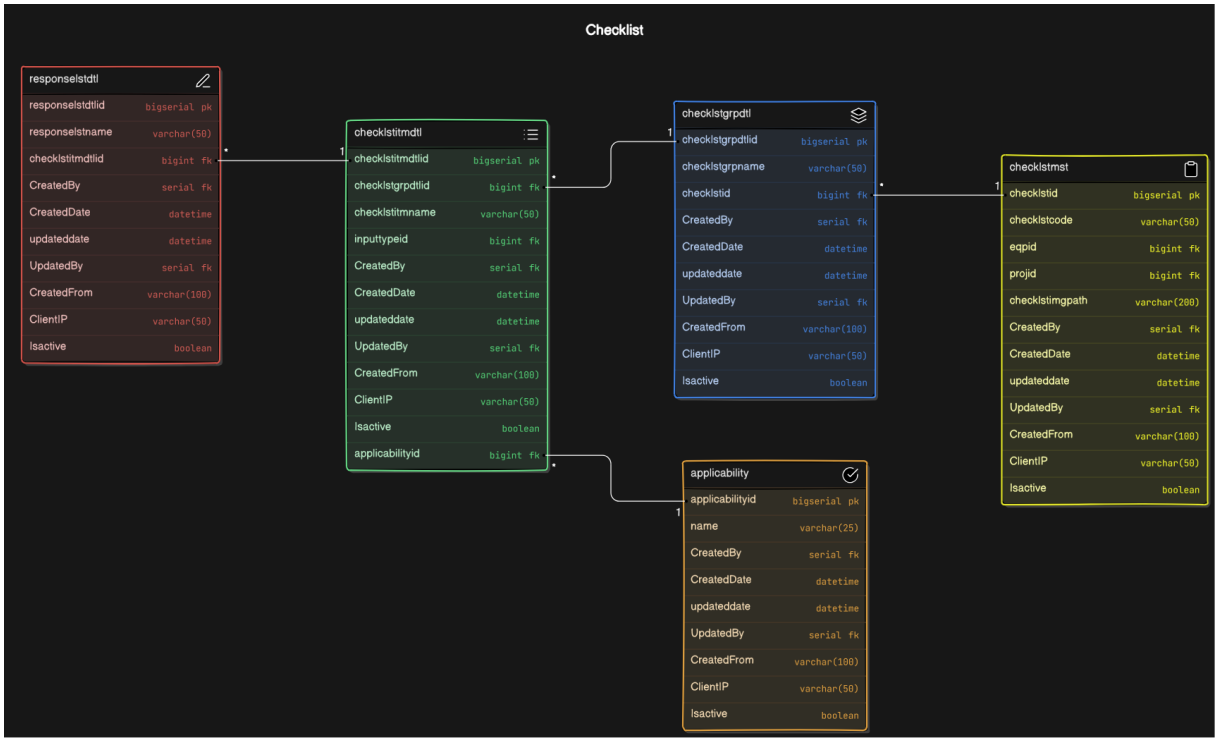


3.2. Entity-Relationship diagram

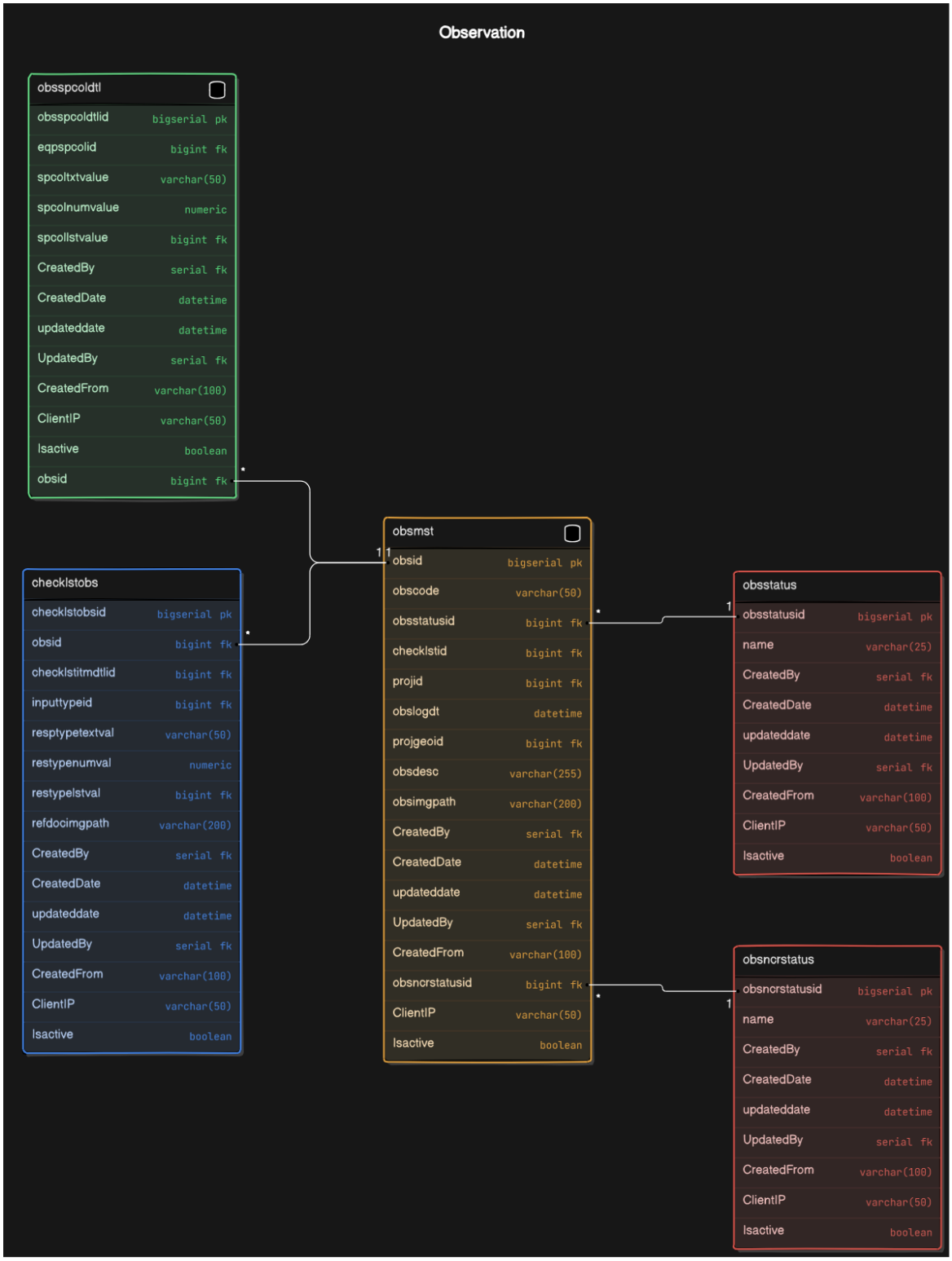
3.2.1. Material Request

3.2.2. Safety-Activty

3.2.3. Checklist



3.2.4. Observation



3.2.5. NCR


NCR
ncrmst
ncrid bigserial pk
ncrcode varchar(50)
obsid bigint fk
projid bigint fk
ncrissdt datetime
projgeoid bigint fk
ncrtypeid bigint fk
deptid bigint fk
audid bigint fk
drno varchar(150)
ncrdesc varchar(100)
recmnds varchar(200)
remedial varchar(200)
ncclsdtl varchar(200)
ncrclsrmks varchar(255)
ncrassnto bigint fk
ncrtrgtdt datetime
ncrclsdt datetime
ncrstatusid bigint fk
CreatedBy serial fk
CreatedDate datetime
updateddate datetime
UpdatedBy serial fk
CreatedFrom varchar(100)
ClientIP varchar(50)
lsactive boolean
*
1
ncrtype
ncrtypeid bigserial pk
name varchar(25)
CreatedBy serial fk
CreatedDate datetime
updateddate datetime
UpdatedBy serial fk
CreatedFrom varchar(100)
ClientIP varchar(50)
lsactive boolean

3.3. Data Dictionary

3.3.1. Material Request

Table Name	matreqmst			**Element Code**	
Purpose of	Store All Material Requisition Records			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
mrid	N	bigserial			PK, auto generated int number
mrcode	N	varchar	50		A Code for all MRIS in order-wise
projid	N	bigserial		FK	Project
projsectionid	Y	bigserial		FK	ProjSectionmst
mrdt	N	datetime			
location	Y	varchar			Location On site if there is one
catid	N	int			catmst
mrfortypeid	N	bigserial		FK	mrfortype (To store a reference to for whom requisition is created either Party or employee or asset or employee or other)
vendorid	Y	bigserial		FK	vendormst
empid	Y	bigserial		FK	UserProfile
assetid	Y	bigserial		FK	assmst
desc	Y	varchar	100		Description for Material Requisition Type Others
ApprovalStatus	N	serial			ApprovalStatusmst
DocStatus	N	boolean			DocStatusmst
mrstatus	N	serial			matreqstatus
remarks	Y	varchar	255		Remarks regarding Material Requisition
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	matreqitmdtl			Element Code	
Purpose of	Store All Material Requisition Item Records			Revision No	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
mritmid	N	bigserial			PK, auto generated int number
mrid	N	bigserial		FK	MaterialRequistionmst
itmid	N	bigserial		FK	Itemmst
reqqty	N	bigserial			Quantity in Number
reqondt	N	datetime			Date Item is required on
uomid	N	bigserial		FK	procurment.uommst
woid	Y	bigserial		FK	subcontract.womst
remarks	Y	varchar	255		Remarks regarding Material Requisition
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	mrfortype			Element Code	
Purpose of	Store All Material Requisition Entity Types			Revision No	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
mrentityid	N	bigserial			PK, auto generated int number
name	N	varchar	50	FK	Entity Name
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

3.3.2. Equipment

Table Name	eqpmst			Element Code	
Purpose of	Store All Equipments Records			Revision No	1
Column Name	Optional	Data Type	Length	Key Reference	Description
eqpid	N	bigserial		PK	PK, auto generated int number
eqpcode	N	varchar	50		Equipment Code
eqpname	N	varchar	50		Equipment Name
projid	N	bigint		FK	admin.project
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	eqpspcoldtl			Element Code	
Purpose of	Store all special columns for each equipment			Revision No	1
Column Name	Optional	Data Type	Length	Key Reference	Description
eqpspcoldtlid	N	bigserial		PK	PK, auto generated int number
eqpspcolname	N	varchar	50		Special Column Label
eqpid	N	bigint		FK	eqpmst
inputtypeid	N	bigint		FK	inputtype
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	inputlistdtl			Element Code	
Purpose of	List of input type for special column inputlistdtl			Revision No	1
Column Name	Optional	Data Type	Length	Key Reference	Description
inputlistdtlid	N	bigserial		PK	PK, auto generated int number
inputlistname	N	varchar	50		Input List name
eqpspcoldtlid	N	bigint		FK	eqpspcol
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	inputtype			Element Code	
Purpose of	Input type for special column labels			Revision No	1
Column Name	Optional	Data Type	Length	Key Reference	Description
inputtypeid	N	bigserial		PK	PK, auto generated int number
name	N	varchar	25		Inputy type name
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

3.3.3. Checklist

Table Name	checklstmst			**Element Code**	
Purpose of	Store All Checklists Records			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
checklstid	N	bigserial		PK	PK, auto generated int number
checklstcode	N	varchar	50		Checklist Code
eqpid	N	bigint		FK	eqpmst
projid	N	bigint		FK	admin.project
checklstimgpath	Y	varchar	200		Checklist Image
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	checklstgrpdtl			**Element Code**	
Purpose of	Store all checklist groups for each checklist			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
checklstgrpdtlid	N	bigserial		PK	PK, auto generated int number
checklstgrpname	N	varchar	50		Check List Group name
checklstid	N	bigint		FK	checklstmst
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	checklstitmdtl			Element Code	
Purpose of	Store all checklist group items for each checklist			Revision No	1
Column Name	Optional	Data Type	Length	Key Reference	Description
checklstitmdtlid	N	bigserial		PK	PK, auto generated int number
checklstitmname	N	varchar	50		Check List Group name
checklstgrpdtlid	N	bigint		FK	checklstgrpdtl
inputtypeid	N	bigint		FK	inputtype
applicabilityid	N	bigint		FK	applicability
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	responselstdtl			Element Code	
Purpose of	List of response type for response type in checklist			Revision No	1
Column Name	Optional	Data Type	Length	Key Reference	Description
responselstdtlid	N	bigserial		PK	PK, auto generated int number
responselstname	N	varchar	50		Response type List name
checklstitmdtlid	N	bigint		FK	checklstitmdtl
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	applicability			**Element Code**	
Purpose of	Applicability of each checklist item			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
applicabilityid	N	bigserial		PK	PK, auto generated int number
name	N	varchar	25		Applicability type name
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

3.3.4. Observation

Table Name	obsmst			**Element Code**	
Purpose of	Store All observation Records			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
obsid	N	bigserial		PK	PK, auto generated int number
obscode	N	varchar	50		Observation Code
checklstid	N	bigint		FK	checklstmst
projid	N	bigint		FK	Projectmst
obslogdt	N	datetime			Observation Log date
projgeoid	N	bigint		FK	Project Geo Table from planning
obsdesc	Y	varchar	255		Observations description
obsstatusid	N	bigint		FK	obsstatus
obsimgpath	Y	varchar	200		Observation Image
obsncrstatusid	Y	bigint		FK	obsncrstatus
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	obsspcoldtl			**Element Code**	
Purpose of	store Special columns values			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
obsspcoldtlid	N	bigserial		PK	PK, auto generated int number
eqpspcolid	N	bigint		FK	eqpspcol
spcoltxtvalue	Y	varchar	50		Value of Special column
spcolnumvalue	Y	numeric			
spcollstvalue	Y	bigint		FK	inputlistid
obsid	N	bigint		FK	obsmst
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			

UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	checklstobs			**Element Code**	
Purpose of	Observations for checklist items			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
checklstobsid	N	bigserial		PK	PK, auto generated int number
checklstitmdtlid	N	bigint		FK	checklstitmdtl
obsid	N	bigint		FK	obsmst
inputtypeid	N	bigint		FK	inputtype
resptypetextval	Y	varchar	50		Text value of if reponsetype text
restypenumval	Y	numeric			Numeric value of if reponsetype numeric
restypelstval	Y	bigint		FK	responselst
refdocimgpath	Y	varchar	200		Image for observation checklist item
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	obsstatus			**Element Code**	
Purpose of	Observation status			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
obsstatusid	N	bigserial		PK	PK, auto generated int number
name	N	varchar	25		Observation status name
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile

CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	obsncrstatus			**Element Code**	
Purpose of	Observation status if obs close with ncr			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
obsncrstatusid	N	bigserial		PK	PK, auto generated int number
name	N	varchar	25		Observation status name
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

3.3.5. NCR

Table Name	ncrmst			Element Code	
Purpose of	Store All ncr Records			Revision No	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
ncrid	N	bigserial		PK	PK, auto generated int number
ncrcode	N	varchar	50		Observation Code
obsid	Y	bigint		FK	obsmst
projid	N	bigint		Fk	projectmst
ncrissdt	N	datetime			NCR Issue date
projgeoid	N	bigint		FK	Project Geo Table from planning
ncrtypeid	N	bigint		FK	ncrtype
deptid	Y	bigint		FK	Department Table
audid	Y	bigint		FK	Audit Table
drno	Y	varchar	150		Checklist code if for observation else text
ncrdesc	Y	varchar	100		NCR Description
recmnds	Y	varchar	200		Recommended actions
remedial	Y	varchar	200		Remedial actions taken
ncclsdtl	Y	varchar	200		Non Conformity Clouse out details
ncrclsrmks	Y	varchar	255		NCR Closing Reamrks
ncrassnto	N	bigint	50	FK	UserProfile
ncrtrgtdt	N	datetime			NCR Target Date
ncrclsdt	N	datetime			NCR Closed date
ncrstatusid	N	bigint		FK	ncrstatus
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Table Name	ncrtype			**Element Code**	
Purpose of	NCR based on			**Revision No**	1
Column Name	**Optional**	**Data Type**	**Length**	**Key Reference**	**Description**
ncrtypeid	N	bigserial		PK	PK, auto generated int number
name	N	varchar	25		NCR based on name
CreatedBy	N	serial		FK	UserProfile
CreatedDate	N	datetime			
updateddate	N	datetime			
UpdatedBy	N	serial		FK	UserProfile
CreatedFrom	N	varchar	100		MAC of the device
ClientIP	Y	varchar	50		Ipaddress of device
Isactive	N	boolean			

Chapter 4: Results and Discussions

4.1. Results

- **Streamlined Material Request Process:** Implemented a user-friendly interface for material requests, reducing time spent on paperwork and improving efficiency in material procurement.

- **Enhanced Safety Measures:** Developed a comprehensive Safety Module covering Equipment Management, Checklist Creation, Observation Recording, and Non-Conformance Reporting (NCR), fostering a culture of safety consciousness throughout the organization.

- **Improved Compliance:** Ensured compliance with industry standards and regulations by integrating safety protocols and checklists into daily operations.

- **Centralized Data Management:** Established a centralized repository for safety-related data, enabling easy access to equipment records, checklists, observations, and NCRs for analysis and decision-making.

- **Reduced Downtime:** Minimized downtime caused by safety incidents through proactive identification of potential hazards, timely equipment inspections, and corrective actions based on observations and NCRs.

- **Improved Communication and Collaboration:** Facilitated seamless communication and collaboration among project teams, supervisors, and safety personnel through integrated tools for sharing information, assigning tasks, and documenting safety-related activities.

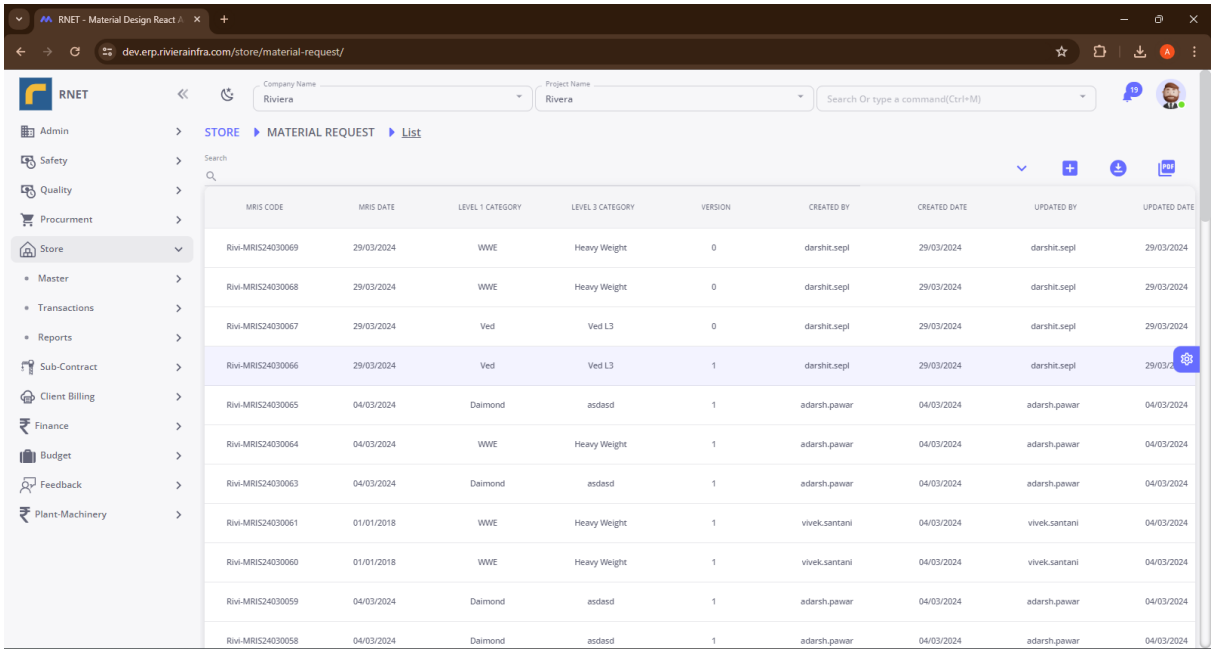
- **Overall Efficiency and Cost Savings:** Achieved overall improvement in operational efficiency, risk mitigation, and cost savings by implementing the Material Request and Safety Module components within the ERP solution, contributing to the company's bottom line.

Chapter 5: User Manual

5.1. List and Form pages with description

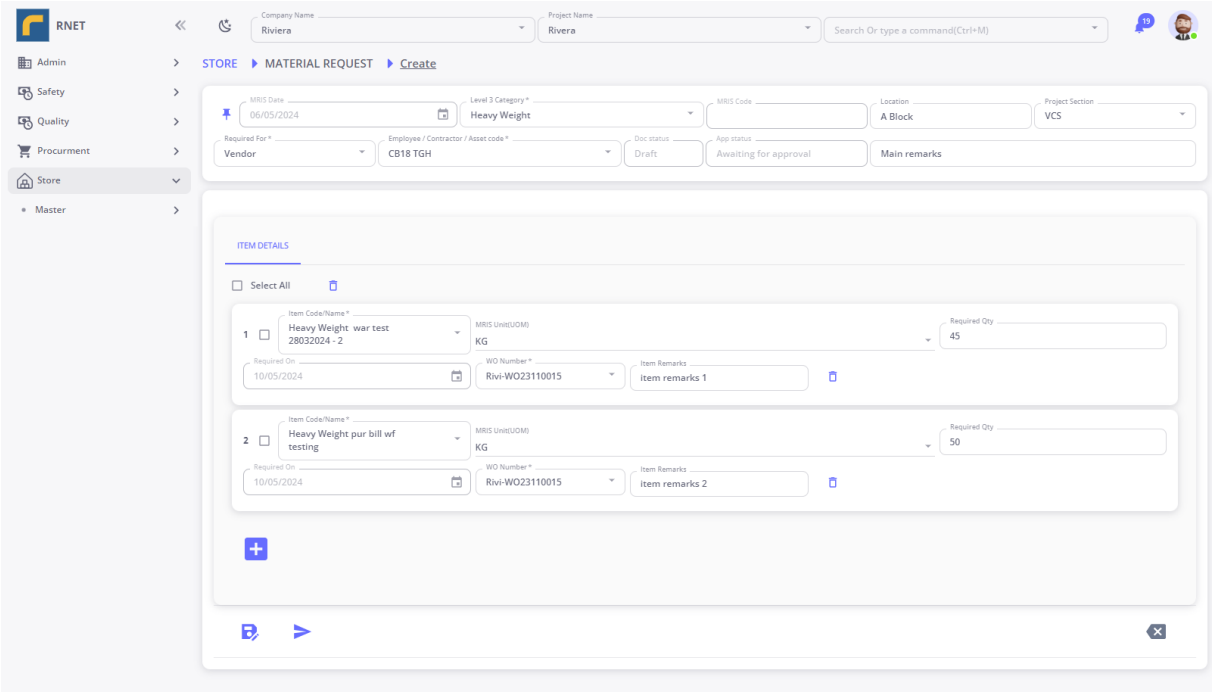
5.1.1. Material Request:

5.1.1.1. List:



- The Material Request component in the store module has a list page that displays the list of all the material requests that have been created in the current company and projects selected from the header.
- The user can apply various filters in the list such as filter approved-disapproved and active-inactive material requests.
- The user can also click on the Edit button to edit the request created if it has the rights or can view the details of the request and also delete the material request if permitted.

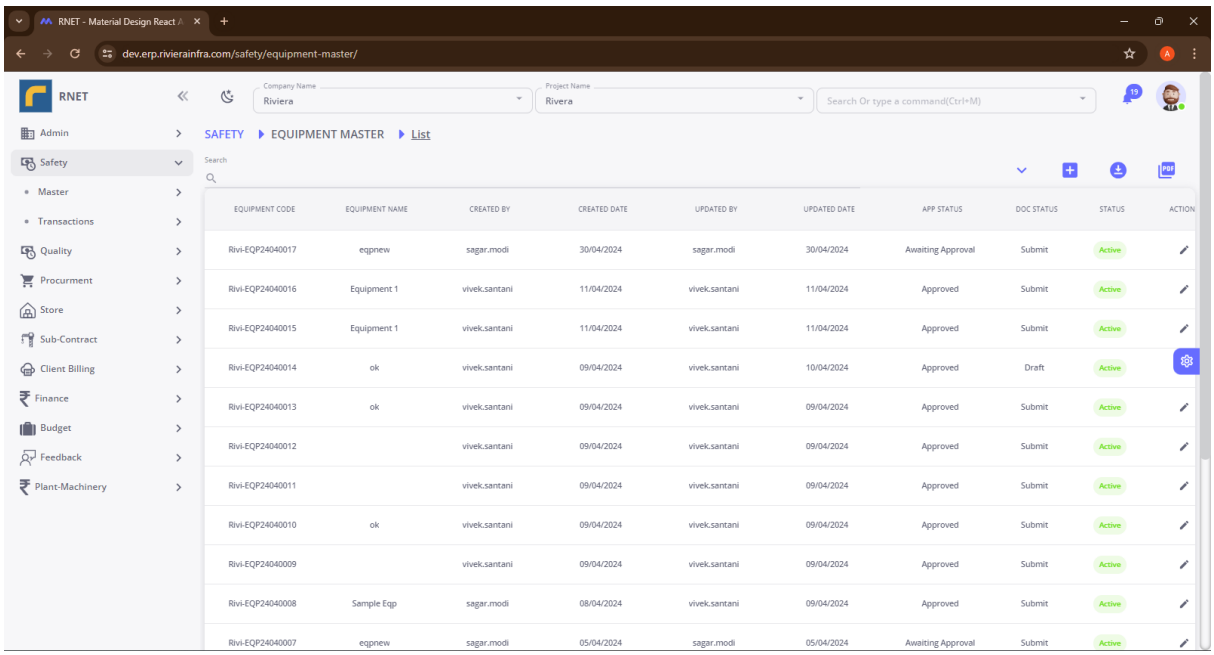
5.1.1.2. Insert or Edit Material Request:



- The detail page is a form page used to create new material requests or edit existing material requests.
- The user can create material requests for vendor, project, asset or any other needs for the project.
- The request would be created for any specific Level 3 category items.
- The items can be selected and the required quantity can be mentioned with the date when the items are required.
- This is a workflow-enabled component so before the store manager can see the issue of this material request it has to be approved fully as per workflow role set to the page.

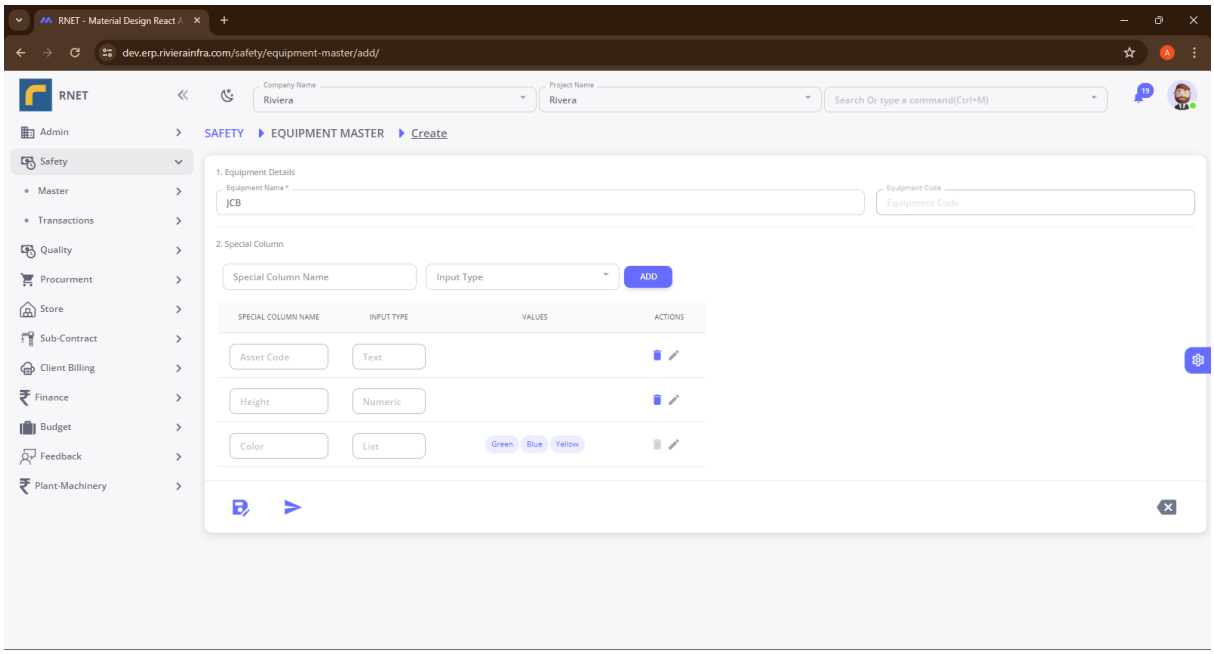
5.1.2. Safety Module

5.1.2.1. Equipment List



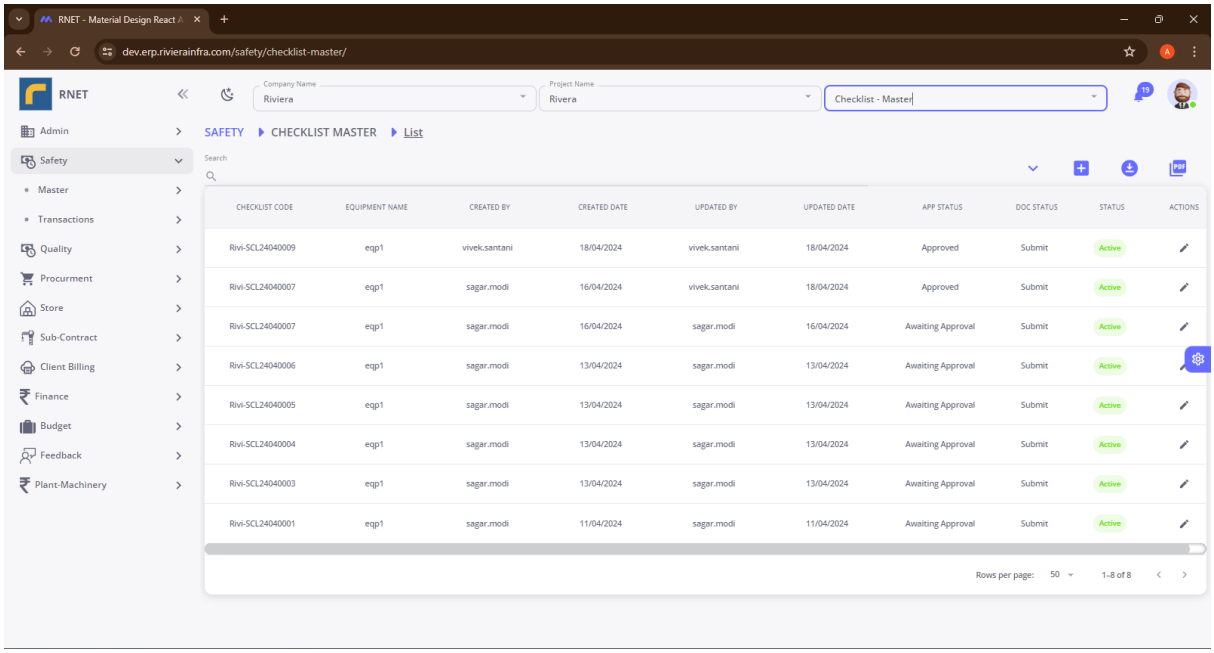
- The Equipment component in the safety module is a master page and has a list page that displays the list of all the Equipment that has been registered in the current company and projects selected from the header.
- The user can apply various filters in the list such as filter approved-disapproved and active-inactive equipment.
- The user can also click on the Edit button to edit the checklist created if it has the rights or can view the details of the equipment and also delete the equipment records if permitted.

5.1.2.2. Insert or Edit Equipment:



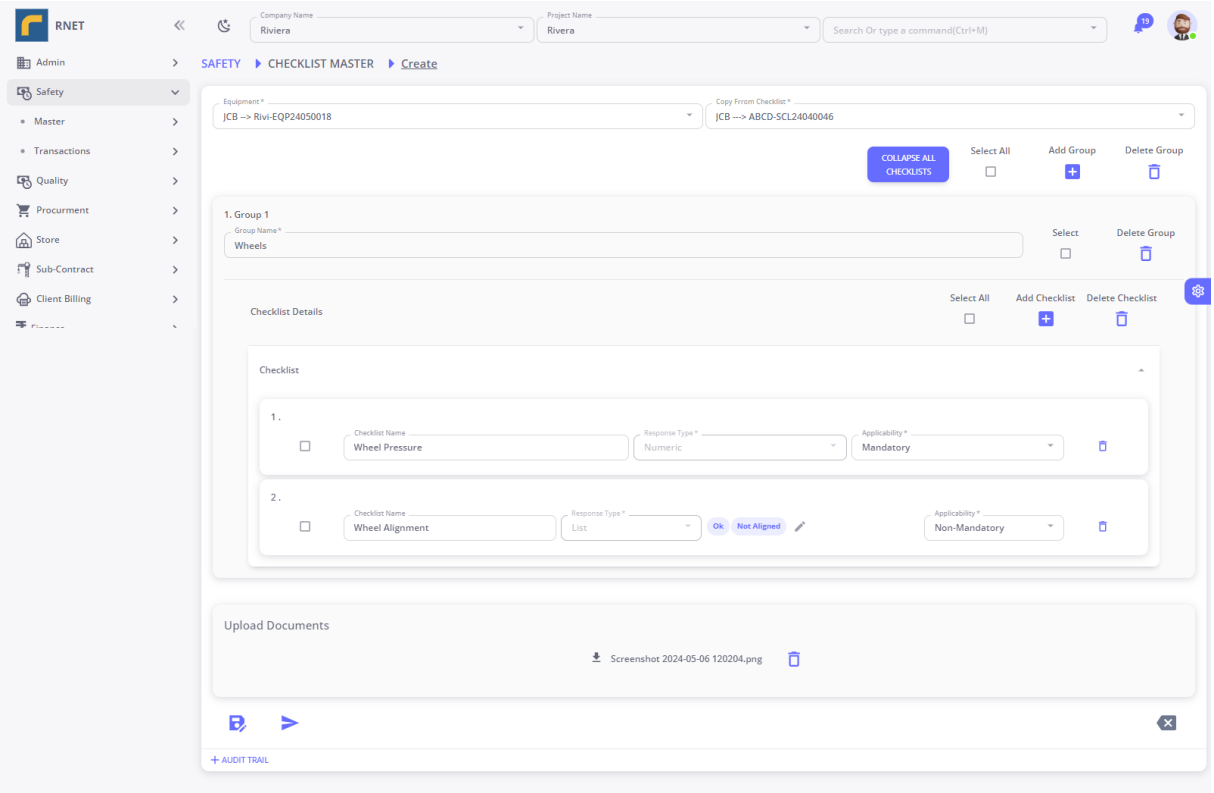
- The detail page is a form page used to create new equipment or edit existing equipment for safety checks.
- The user can register equipment that would be used in the project and can create custom special columns for specific equipment that are needed to be taken into consideration while taking an observation.
- The special column can have a predefined type for the observation to be noted such as Text, Numeric or List.
- For list type the user can also set parameters to select from while taking observation.

5.1.2.3. Checklist List



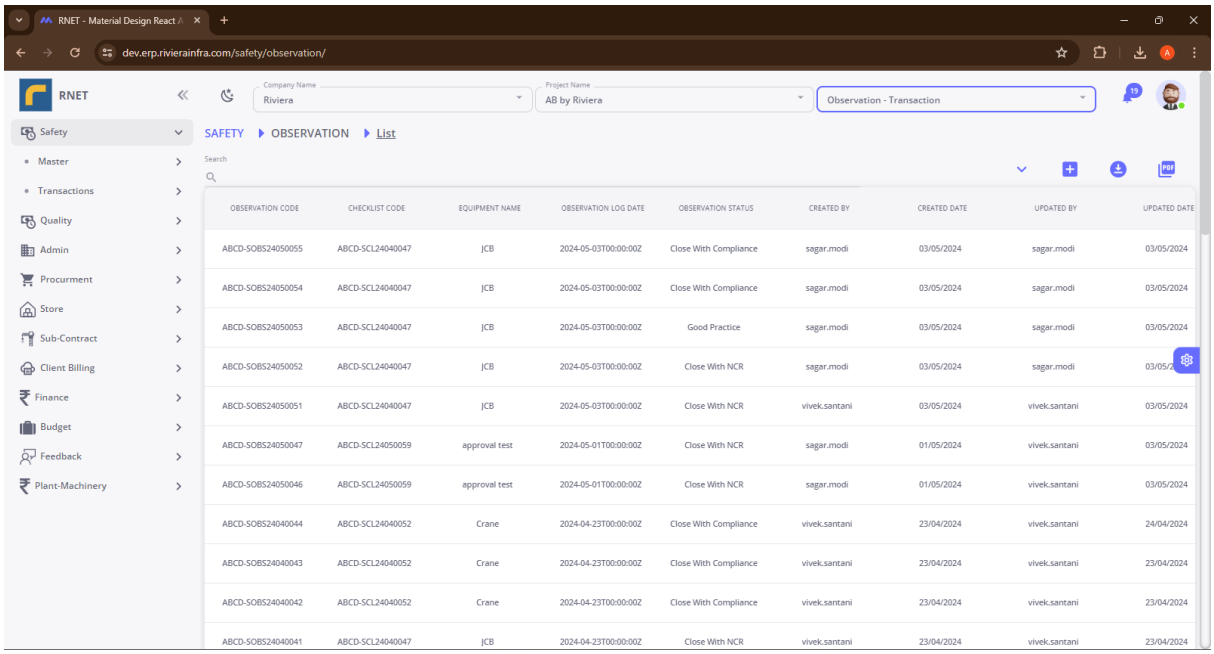
- The Checklist component in the safety module is a master page and has a list page that displays the list of all the Checklists that have been created for specific equipment in the current company and projects selected from the header.
- The user can apply various filters in the list such as filter approved-disapproved and active-inactive checklists.
- The user can also click on the Edit button to edit the checklist created if it has the rights or can view the details of the checklist and also delete the checklist records if permitted.

5.1.2.4. Insert or Edit Checklist:



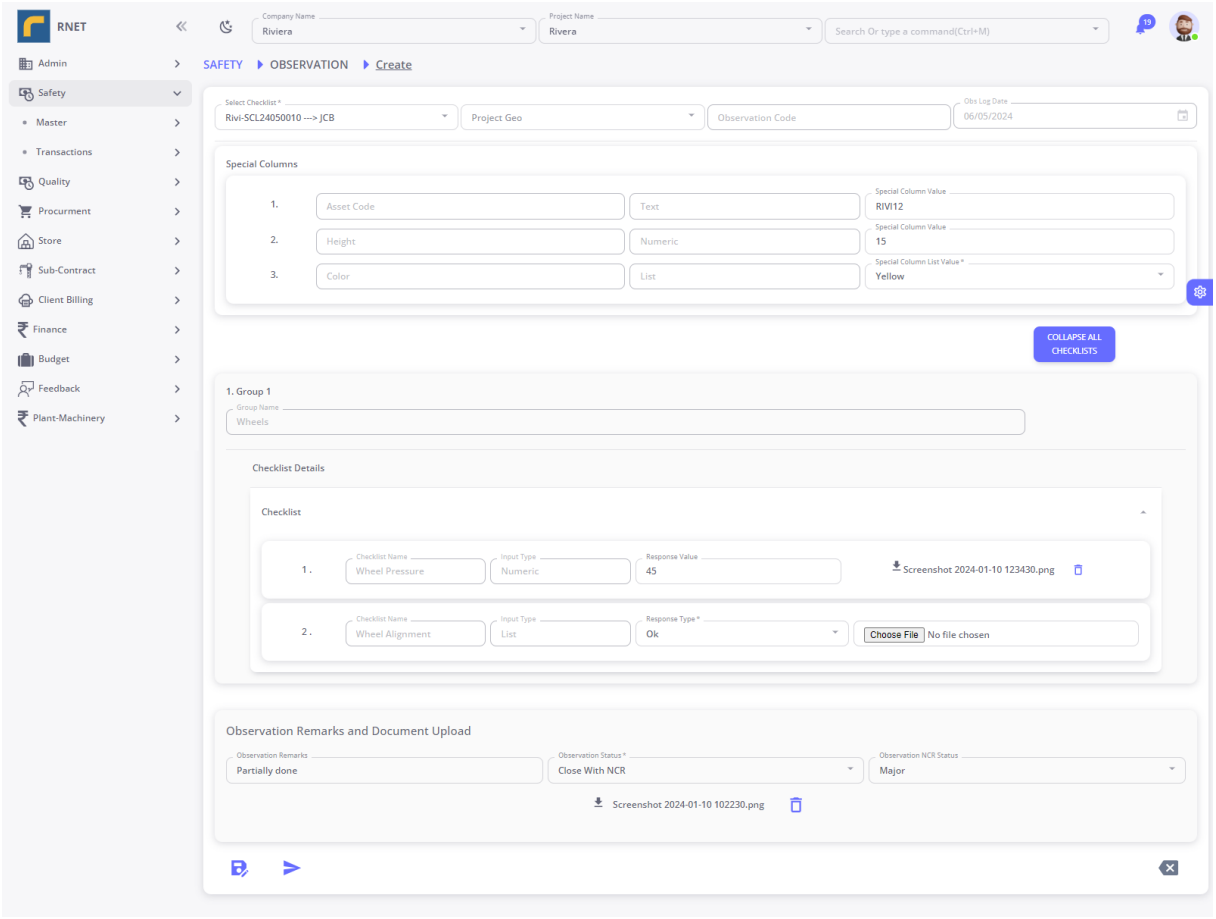
- The detail page is a form page used to create new checklists or edit existing checklists created for specific equipment for safety checks.
- The user can create checklists in groups for specific equipment that would be used in the project for its safety check and can tag a checklist to be mandatory or non-mandatory to check-in observation.
- The checklist can have a predefined type for the observation to be noted such as Text, Numeric or List.
- For list type the user can also set parameters to select from while taking observation.

5.1.2.5. Observation List



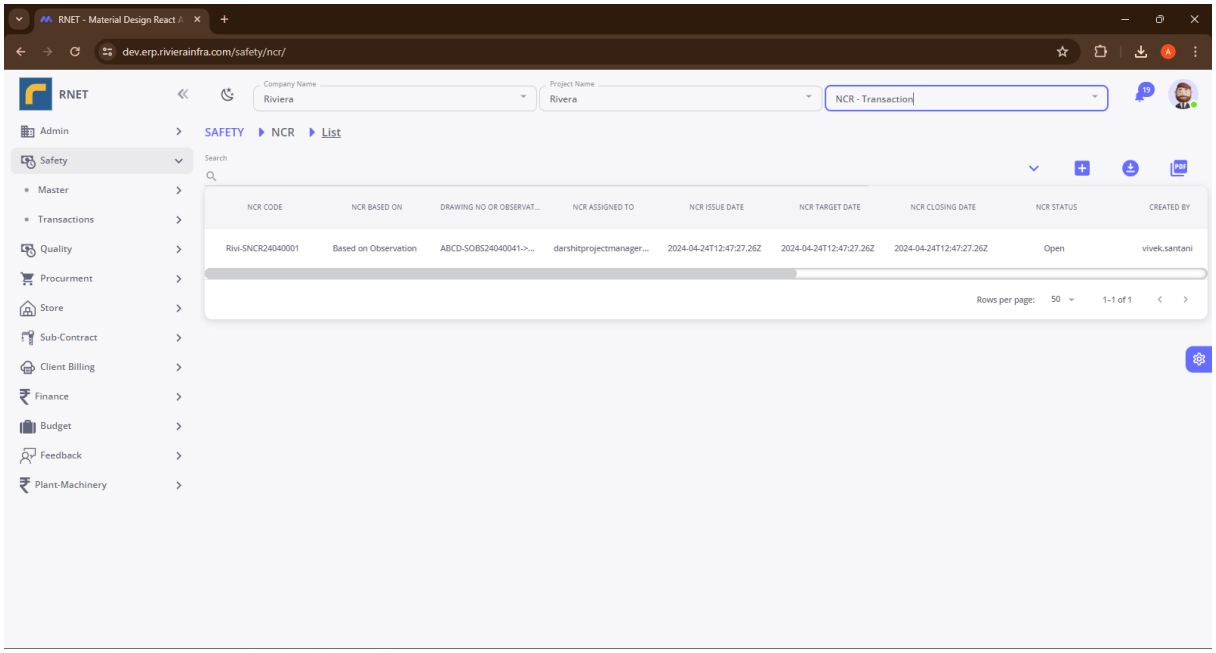
- The Observation component in the safety module is a transaction page and has a list page that displays the list of all the Observations that have been created for specific checklists in the current company and projects selected from the header.
- The user can apply various filters in the list such as filter approved-disapproved and active-inactive observations.
- The user can also click on the Edit button to edit the observation created if it has the rights or can view the details of the observation and also delete the observation records if permitted.

5.1.2.6. Insert or Edit Observation:



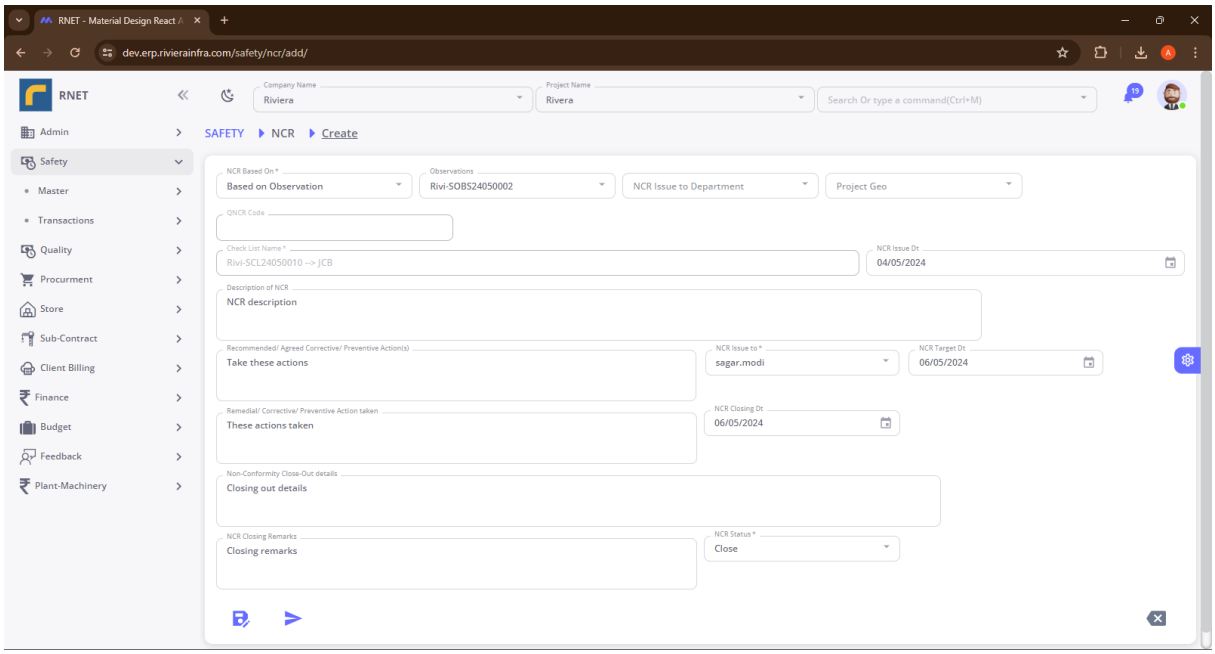
- The detail page is a form page used to create new observations or edit existing observations created for specific checklist for safety checks.
- The user can create observation by selecting a checklist from the project for equipment and has to enter all the required data by checking the safety of the equipment.
- This is a workflow-enabled component so all the observations will go through the workflow as per the type of role assigned before being fully approved and till then equipment at the site will not be considered as safe to use.

5.1.2.7. NCR List



- The NCR component in the safety module is a transaction page and has a list page that displays the list of all the NCRs that have been created for specific observation in the current company and projects selected from the header.
- The user can apply various filters in the list such as filter approved-disapproved and active-inactive observations.
- The user can also click on the Edit button to edit the NCR created if it has the rights or can view the details of the NCR and also delete the NCR records if permitted.

5.1.2.8. Insert or Edit Observation:



- The detail page is a form page used to create new NCRs or edit existing NCRs created for specific observations for safety checks.
- The user can create NCR based on observation, audit, direct or any client complaint.
- An NCR can be assigned to a user with a target date and a close date with recommended actions that can be taken.
- This is a workflow-enabled component so all the NCR will go through the workflow as per the type of role assigned before being fully approved.

Chapter 6: Testing

6.1. Material Request

6.1.1. Material Request List Page

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_001	Verify MRIS listing page	Store – MRIS-ListPage	User should be logged in with proper permissions	1	Launch the application and navigate to the MRIS listing page	MRIS listing page should load successfully with all the required filters and data grid	MRIS listing page loads successfully with all required filters and data grid.
				2	Apply different filters (from date, to date, approval status, document status, status)	Data grid should display filtered data correctly	MRIS listing page loads successfully with all required filters and data grid.
				3	Click on the "Add" button	User should be redirected to the MRIS add/edit page	User is redirected back to the listing page after creating/updating MRIS.
TC_002	Verify MRIS add/edit functionality	Store – MRIS-ListPage	User should be logged in with proper permissions	1	Navigate to the MRIS add/edit page	MRIS add/edit form should load successfully	MRIS add/edit form loads successfully.
				2	Fill in the required fields and submit the form	MRIS should be created/updated successfully, and the user should be redirected back to the listing page	MRIS is created/updated successfully upon submitting the form.
				3	Verify the newly created/updated MRIS in the listing page	The new/updated MRIS should be visible in the listing page	The new/updated MRIS is visible in the listing page.

TC_003	Verify MRIS delete functionality	Store – MRIS-ListPage	User should be logged in with proper permissions and have at least one MRIS record	1	Navigate to the MRIS listing page	MRIS listing page should load successfully with at least one record	MRIS listing page loads successfully with at least one record.
				2	Select a MRIS record and click the "Delete" action	A confirmation dialog should appear if the record is not in workflow.	A confirmation dialog appears upon selecting a record and clicking "Delete".
				3	Confirm the delete action	The selected MRIS record should be deleted, and the data grid should refresh without the deleted record	The selected MRIS record is deleted, and the data grid refreshes without the deleted record.
TC_004	Verify search functionality	Store – MRIS-ListPage	User should be logged in with proper permissions and have multiple MRIS records	1	Navigate to the MRIS listing page	MRIS listing page should load successfully with multiple records	MRIS listing page loads successfully with all required filters and data grid
				2	Enter a search term in the search input field	Data grid should display only the records matching the search term	Data grid displays only the records matching the search term entered in the search input field.
				3	Clear the search input field	Data grid should display all records again	MRIS listing page loads successfully with all required filters and data grid
TC_005	Verify export functionality	Store – MRIS-ListPage	User should be logged in with proper permissions and have MRIS records.	1	Navigate to the MRIS listing page	MRIS listing page should load successfully with records	MRIS listing page loads successfully with all required filters and data grid
				2	Click on the "PDF" or "Excel" export button	The application should generate and download the respective file format with the MRIS data	The application generates and downloads the respective file format with the MRIS data

6.1.2. Material Request Form Page

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_01	Verify the page is opened in add mode and fields are available.	Store – MRIS - FormPage	User should have viewing and adding permission	**1**	Click on add button in MRIS list page.	The MRIS form should be opened with all the fields according to validations.	The MRIS form opens with all the fields according to validations.
				2	Fill the details in form as required validations.	Only selected number of fields should be available based on user's interaction.	Only selected number of fields are available based on user's interaction.
TC_02	Verify the deletion of items based on the checkbox selection	Store – MRIS – FormPage	User should have viewing and adding permission	**1**	Tick the checkboxes which are to be selected along with select.	The items with checked checkboxes should be selected.	The items with checked checkboxes are selected.
				2	Click on delete button for individual item or click on delete button for selected boxes.	The individual item or the items with checked checkboxes should be removed from the form.	The individual item or the items with checked checkboxes are removed from the form.
TC_03	Verify the page is opened in edit mode and only editable fields are available.	Store – MRIS - FormPage	User should have viewing and editing permission	**1**	Click on edit button in MRIS list page for that record.	The form page should be loaded with all the fields filled with the data.	The form page is loaded with all the fields filled with the data.
				2	Update the details in form.	Only the editable fields should be editable.	Only the editable fields are editable.
TC_04	Verify the submission of the form when the data is not filled or invalid.	Store – MRIS - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The alerts and messages should be displayed wherever the fields are bot filled or are invalid.	The alerts and messages are displayed wherever the fields are bot filled or are invalid.
TC_05	Verify the submission of the form when the data is fully filled and valid.	Store – MRIS - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The form should be submitted and the record of it should be displayed on MRIS list page.	The form is submitted and the record of is displayed on MRIS list page.

TC_06	Verify the edit page opened by user when it is in workflow.	Store – MRIS - FormPage	User should have viewing permission	**1**	Click on edit button in MRIS list page for that record.	The form page should be loaded with all the fields filled with the data	The form page is loaded with all the fields filled with the data.
				2	Scroll down to see the action button visibility.	The submission buttons should not be visible so that user cannot edit data.	The submission buttons are not visible so that user cannot edit data.
TC_07	Verify the edit page opened by approver when it is in workflow.	Store – MRIS - FormPage	User should have viewing permission	**1**	Click on edit button in MRIS list page for that record.	The form page should be loaded with all the fields filled with the data	The form page is loaded with all the fields filled with the data.
				2	Click on approve or disapprove button.	Only the approval/dis approval button should be visible and on clicking the record must be approved or disapproved.	Only the approval/dis approval button is visible and on clicking the record is approved or disapproved.

6.1.3. Material Request Backend

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Test Steps	Expected Result	Actual Result
TC_01	Verify GetMaterialRequestForList API	Store – MRIS – List Page	User should be logged in with proper permissions	Call the API GET /api/MaterialRequest/GetMaterialRequestForList with valid data	MRIS listing should be successfully retrieved with all required filters and data grid	MRIS listing was successfully retrieved with all required filters and data grid as expected.
TC_02	Verify GetMaterialRequestById API	Store – MRIS-FormPage	User should be logged in with proper permissions	Call the API GET /api/MaterialRequest/GetMaterialRequestById with valid data	Material Request details should be successfully retrieved by ID	Material Request details were successfully retrieved by ID as expected.
TC_03	Verify InsertMaterialRequest API	Store – MRIS-FormPage	User should be logged in with proper permissions	Call the API POST /api/MaterialRequest/InsertMaterialRequest with valid data	Material Request should be successfully inserted	Material Request was successfully inserted as expected.
TC_04	Verify UpdateMaterialRequest API	Store – MRIS-FormPage	User should be logged in with proper permissions	Call the API PUT /api/MaterialRequest/UpdateMaterialRequest with valid data	Material Request should be successfully updated	Material Request was successfully updated as expected.
TC_05	Verify UpdateMRApproveDisapprove API	Store – MRIS-Formpage	User should be logged in with proper permissions	Call the API POST /api/MaterialRequest/UpdateMRApproveDisapprove with valid data	Material Request should be successfully approved/disapproved	Material Request was successfully approved/disapproved as expected.
TC_006	Verify GetStoreApprovalAuditList API	Store - Approval	User should be logged in with proper permissions	Call the API GET /api/MaterialRequest/GetStoreApprovalAuditList with valid data	Approval audit list should be successfully retrieved with specified program ID and transaction ID	Approval audit list was successfully retrieved with specified program ID and transaction ID as expected.
TC_007	Verify DeleteStoreTransaction API	Store - Transaction	User should be logged in with proper permissions	Call the API PUT /api/MaterialRequest/DeleteStoreTransaction with valid data	Store transaction should be successfully deleted based on the provided data	Store transaction was successfully deleted based on the provided data as expected.
TC_008	Verify GetMaterialRequiredForType API	Store - Material	-	Call the API GET /api/MaterialRequest/GetMaterialRequiredForType	Material requirements for type should be successfully retrieved	Material requirements for type were successfully retrieved as expected.

6.2. Safety Module

6.2.1. Safety-Activity

6.2.1.1. Safety-Activity List Page

Test Case Id	Test Title/Test Case Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_001	Verify safety activity listing page	Safety – Safety Activity - ListPage	User should be logged in with proper permissions	1	Launch the application and navigate to the safety activity listing page	Safety-activity listing page should load successfully with all the required filters and data grid	Safety-activity listing page loads successfully with all required filters and data grid.
				2	Apply different filters (from date, to date, approval status, document status, status)	Data grid should display filtered data correctly	Safety-activity listing page loads successfully with all required filters and data grid.
				3	Click on the "Add" button	The user should be redirected to the Safety-activity add/edit page	The user is redirected to the Safety-activity add/edit page
TC_002	Verify Safety-activity add/edit functionality	Safety – Safety Activity - ListPage	User should be logged in with proper permissions	1	Navigate to the Safety-Activity add/edit page	Safety-Activity add/edit form should load successfully	Safety-Activity add/edit form loads successfully.
				2	Fill in the required fields and submit the form	Safety-Activity should be created/updated successfully, and the user should be redirected back to the listing page	Safety-Activity is created/updated successfully upon submitting the form.
				3	Verify the newly created/updated Safety-Activity in the listing page	The new/updated Safety-Activity should be visible in the listing page	The new/updated Safety-Activity is visible in the listing page.
TC_	Verify	Safety	User	1	Navigate to	Safety-Activity	Safety-Activity

003	Safety-Activity delete functionality	– Safety Activity - ListPage	should be logged in with proper permissions and have at least one Safety-Activity record		the Safety-Activity listing page	listing page should load successfully with at least one record	listing page loads successfully with at least one record.
				2	Select a Safety-Activity record and click the "Delete" action	A confirmation dialog should appear if the record is not in workflow.	A confirmation dialog appears upon selecting a record and clicking "Delete".
				3	Confirm the delete action	The selected Safety-Activity record should be deleted, and the data grid should refresh without the deleted record	The selected Safety-Activity record is deleted, and the data grid refreshes without the deleted record.
TC_004	Verify search functionality	Safety – Safety Activity - ListPage	User should be logged in with proper permissions and have multiple Safety-Activity records	1	Navigate to the Safety-Activity listing page	Safety-Activity listing page should load successfully with multiple records	Safety-Activity listing page loads successfully with all required filters and data grid
				2	Enter a search term in the search input field	Data grid should display only the records matching the search term	Data grid displays only the records matching the search term entered in the search input field.
				3	Clear the search input field	Data grid should display all records again	Safety-Activity listing page loads successfully with all required filters and data grid
TC_005	Verify export functionality	Safety – Safety Activity - ListPage	User should be logged in with proper permissions and have Safety-Activity records.	1	Navigate to the Safety-Activity listing page	Safety-Activity listing page should load successfully with records	Safety-Activity listing page loads successfully with all required filters and data grid
				2	Click on the "PDF" or "Excel" export button	The application should generate and download the respective file format with the Safety-Activity data	The application generates and downloads the respective file format with the Safety-Activity data

6.2.1.2. Safety-Activity Form Page

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_01	Verify the page is opened in add mode and fields are available.	Safety – Safety Activity - FormPage	User should have viewing and adding permission	**1**	Click on add button on Safety-Activity list page.	The Safety-Activity form should be opened with all the fields according to validations.	The Safety-Activity form opens with all the fields according to validations.
				2	Fill the details in form as required validations.	Only selected number of fields should be available based on user's interaction.	Only selected number of fields are available based on user's interaction.
TC_02	Verify the page is opened in edit mode and only editable fields are available.	Safety – Safety Activity - FormPage	User should have viewing and editing permission	**1**	Click on edit button in Safety-Activity list page for that record.	The form page should be loaded with all the fields filled with the data.	The form page is loaded with all the fields filled with the data.
				2	Update the details in form.	Only the editable fields should be editable.	Only the editable fields are editable.
TC_03	Verify the submission of the form when the data is not filled or invalid.	Safety – Safety Activity - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The alerts and messages should be displayed wherever the fields are bot filled or are invalid.	The alerts and messages are displayed wherever the fields are bot filled or are invalid.
TC_04	Verify the submission of the form when the data is fully filled and valid.	Safety – Safety Activity - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The form should be submitted and the record of it should be displayed on Safety-Activity list page.	The form is submitted and the record of is displayed on Safety-Activity list page.
TC_05	Verify the edit page opened by user when it is in workflow.	Safety – Safety Activity - FormPage	User should have viewing permission	**1**	Click on edit button in Safety-Activity list page for that record.	The form page should be loaded with all the fields filled with the data	The form page is loaded with all the fields filled with the data.
				2	Scroll down to see the action button visibility.	The submission buttons should not be visible so that user cannot edit data.	The submission buttons are not visible so that user cannot edit data.

6.2.1.3. Safety-Activity Backend

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Test Steps	Expected Result	Actual Result
TC_01	Verify GetEquipmentForList API	Safety – Safety Activity – List Page	User should be logged in with proper permissions	Call the API GET /api/Equipment/ GetEquipmentForList with valid data	Safety-Activity listing should be successfully retrieved with all required filters and data grid	Safety-Activity listing was successfully retrieved with all required filters and data grid as expected.
TC_02	Verify GetEquipmentById API	Safety – Safety Activity-FormPage	User should be logged in with proper permissions	Call the API GET /api/Equipment/ GetEquipmentById with valid data	Safety Activity details should be successfully retrieved by ID	Safety Activity details were successfully retrieved by ID as expected.
TC_03	Verify InsertEquipment API	Safety – Safety Activity FormPage	User should be logged in with proper permissions	Call the API POST /api/Equipment/ InsertEquipment with valid data	Safety Activity should be successfully inserted	Safety Activity was successfully inserted as expected.
TC_04	Verify UpdateEquipment API	Safety – Safety Activity-FormPage	User should be logged in with proper permissions	Call the API PUT /api/Equipment/UpdateEquipment with valid data	Safety Activity should be successfully updated	Safety Activity was successfully updated as expected.
TC_05	Verify DeleteSafetyTransaction API	Safety - Transaction	User should be logged in with proper permissions	Call the API PUT /api/DeleteSafety Transaction with valid data	Safety transaction should be successfully deleted based on the provided data	Safety transaction was successfully deleted based on the provided data as expected.
TC_06	Verify GetInputtype API	Safety – Safety Equipment	-	Call the API GET /api/Equipment / GetInputtype	Inputype list should be successfully retrieved	Inputype list was successfully retrieved as expected.

6.2.2. Checklist

6.2.2.1. Safety-Checklist List Page

Test Case Id	Test Title/Test Case Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_001	Verify Checklist listing page	Safety – Checklist - ListPage	User should be logged in with proper permissions	1	Launch the application and navigate to the Checklist listing page	Safety-Checklist listing page should load successfully with all the required filters and data grid	Safety-Checklist listing page loads successfully with all required filters and data grid.
				2	Apply different filters (from date, to date, approval status, document status, status)	Data grid should display filtered data correctly	Safety-Checklist listing page loads successfully with all required filters and data grid.
				3	Click on the "Add" button	The user should be redirected to the Safety-Checklist add/edit page	The user is redirected to the Safety-Checklist add/edit page
TC_002	Verify Safety-Checklist add/edit functionality	Safety – Checklist - ListPage	User should be logged in with proper permissions	1	Navigate to the Safety-Checklist add/edit page	Safety-Checklist add/edit form should load successfully	Safety-Checklist add/edit form loads successfully.
				2	Fill in the required fields and submit the form	Safety-Checklist should be created/updated successfully, and the user should be redirected back to the listing page	Safety-Checklist is created/updated successfully upon submitting the form.
				3	Verify the newly created/updated Safety-Checklist in the listing page	The new/updated Safety-Checklist should be visible in the listing page	The new/updated Safety-Checklist is visible in the listing page.
TC_003	Verify Safety-Checklist delete functionality	Safety – Checklist - ListPage	User should be logged in with proper permissions and have	1	Navigate to the Safety-Checklist listing page	Safety-Checklist listing page should load successfully with at least one record	Safety-Checklist listing page loads successfully with at least one record.
				2	Select a Safety-	A confirmation dialog should	A confirmation dialog appears upon

			at least one Safety-Checklist record		Checklist record and click the "Delete" action	appear if the record is not in workflow.	selecting a record and clicking "Delete".
				3	Confirm the delete action	The selected Safety-Checklist record should be deleted, and the data grid should refresh without the deleted record	The selected Safety-Checklist record is deleted, and the data grid refreshes without the deleted record.
TC_004	Verify search functionality	Safety – Checklist - ListPage	User should be logged in with proper permissions and have multiple Safety-Checklist records	1	Navigate to the Safety-Checklist listing page	Safety-Checklist listing page should load successfully with multiple records	Safety-Checklist listing page loads successfully with all required filters and data grid
				2	Enter a search term in the search input field	Data grid should display only the records matching the search term	Data grid displays only the records matching the search term entered in the search input field.
				3	Clear the search input field	Data grid should display all records again	Safety-Checklist listing page loads successfully with all required filters and data grid
TC_005	Verify export functionality	Safety – Checklist - ListPage	User should be logged in with proper permissions and have Safety-Checklist records.	1	Navigate to the Safety-Checklist listing page	Safety-Checklist listing page should load successfully with records	Safety-Checklist listing page loads successfully with all required filters and data grid
				2	Click on the "PDF" or "Excel" export button	The application should generate and download the respective file format with the Safety-Checklist data	The application generates and downloads the respective file format with the Safety-Checklist data

6.2.2.2. Safety-Checklist Form Page

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_01	Verify the page is opened in add mode and fields are available.	Safety – Checklist - FormPage	User should have viewing and adding permission	**1**	Click on add button on Safety-Checklist list page.	The Safety-Checklist form should be opened with all the fields according to validations.	The Safety-Checklist form opens with all the fields according to validations.
				2	Fill the details in form as required validations.	Only selected number of fields should be available based on user's interaction.	Only selected number of fields are available based on user's interaction.
TC_02	Verify the page is opened in edit mode and only editable fields are available.	Safety – Checklist - FormPage	User should have viewing and editing permission	**1**	Click on edit button in Safety-Checklist list page for that record.	The form page should be loaded with all the fields filled with the data.	The form page is loaded with all the fields filled with the data.
				2	Update the details in form.	Only the editable fields should be editable.	Only the editable fields are editable.
TC_03	Verify the submission of the form when the data is not filled or invalid.	Safety – Checklist - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The alerts and messages should be displayed wherever the fields are bot filled or are invalid.	The alerts and messages are displayed wherever the fields are bot filled or are invalid.
TC_04	Verify the submission of the form when the data is fully filled and valid.	Safety – Checklist - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The form should be submitted and the record of it should be displayed on Safety-Checklist list page.	The form is submitted and the record of is displayed on Safety-Checklist list page.
TC_05	Verify the edit page opened by user when it is in workflow.	Safety – Checklist - FormPage	User should have viewing permission	**1**	Click on edit button in Safety-Checklist list page for that record.	The form page should be loaded with all the fields filled with the data	The form page is loaded with all the fields filled with the data.
				2	Scroll down to see the action button visibility.	The submission buttons should not be visible so that user cannot edit data.	The submission buttons are not visible so that user cannot edit data.
TC_06	Verify the add page when the	Safety – Checklist -	User should have viewing and adding	**1**	Click on add button on Safety-	The Safety-Checklist form should be opened	The Safety-Checklist form opens with all the

	details are copied from other checklist	FormPage	permission		Checklist list page.	with all the fields according to validations.	fields according to validations.
				2	Click on copy from checklist dropdown and select a checklist to be copied.	The checklist page should be filled with copied checklist details.	The checklist page is filled with copied checklist details.

6.2.2.3. Safety-Checklist Backend

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Test Steps	Expected Result	Actual Result
TC_01	Verify GetChecklistForList API	Safety – Checklist – List Page	User should be logged in with proper permissions	Call the API GET /api/Checklist/ GetChecklistForList with valid data	Safety-Checklist listing should be successfully retrieved with all required filters and data grid	Safety-Checklist listing was successfully retrieved with all required filters and data grid as expected.
TC_02	Verify GetChecklistById API	Safety – Checklist-FormPage	User should be logged in with proper permissions	Call the API GET /api/Checklist/ GetChecklistById with valid data	Checklist details should be successfully retrieved by ID	Checklist details were successfully retrieved by ID as expected.
TC_03	Verify InsertChecklist API	Safety – Checklist FormPage	User should be logged in with proper permissions	Call the API POST /api/Checklist/ InsertChecklist with valid data	Checklist should be successfully inserted	Checklist was successfully inserted as expected.
TC_04	Verify UpdateChecklist API	Safety – Checklist-FormPage	User should be logged in with proper permissions	Call the API PUT /api/Checklist/UpdateChecklist with valid data	Checklist should be successfully updated	Checklist was successfully updated as expected.
TC_05	Verify DeleteSafetyTransaction API	Safety - Transaction	User should be logged in with proper permissions	Call the API PUT /api/DeleteSafety Transaction with valid data	Safety transaction should be successfully deleted based on the provided data	Safety transaction was successfully deleted based on the provided data as expected.
TC_06	Verify GetApplicability API	Safety – Safety Checklist	-	Call the API GET /api/Checklist/ GetApplicability	Applicability list should be successfully retrieved	Applicability list was successfully retrieved as expected.
TC_07	Verify GetEquipmentByProjId API	Safety – Safety Checklist	-	Call the API GET /api/Checklist/ GetEquipmentByProjId	List of equipments of the same project should be successfully retrieved	List of equipments of the same project was successfully retrieved as expected.
TC_08	Verify GetChecklistByComid API	Safety – Safety Checklist	-	Call the API GET /api/Checklist/ GetChecklistByComid	List of checklists of the same company should be successfully retrieved	List of checklists of the same company was successfully retrieved as expected.
TC_09	Verify InsertChecklistImage API	Safety – Safety Checklist	User should be logged in with proper permissions	Call the API POST /api/Checklist/ InsertChecklistImage	Checklist image should be successfully inserted	Checklist image was successfully inserted

6.2.3. Observation

6.2.3.1. Safety-Observation List Page

Test Case Id	Test Title/Test Case Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_001	Verify Observation listing page	Safety – Observation - ListPage	User should be logged in with proper permissions	1	Launch the application and navigate to the Observation listing page	Safety-Observation listing page should load successfully with all the required filters and data grid	Safety-Observation listing page loads successfully with all required filters and data grid.
				2	Apply different filters (from date, to date, approval status, document status, status)	Data grid should display filtered data correctly	Safety-Observation listing page loads successfully with all required filters and data grid.
				3	Click on the "Add" button	The user should be redirected to the Safety-Observation add/edit page	The user is redirected to the Safety-Observation add/edit page
TC_002	Verify Safety-Observation add/edit functionality	Safety – Observation - ListPage	User should be logged in with proper permissions	1	Navigate to the Safety-Observation add/edit page	Safety-Observation add/edit form should load successfully	Safety-Observation add/edit form loads successfully.
				2	Fill in the required fields and submit the form	Safety-Observation should be created/updated successfully, and the user should be redirected back to the listing page	Safety-Observation is created/updated successfully upon submitting the form.
				3	Verify the newly created/updated Safety-Observation in the listing page	The new/updated Safety-Observation should be visible in the listing page	The new/updated Safety-Observation is visible in the listing page.
TC_003	Verify Safety-Observation delete	Safety – Observation - ListPage	User should be logged in with proper permissions	1	Navigate to the Safety-Observation listing page	Safety-Observation listing page should load successfully	Safety-Observation listing page loads successfully with at least one record.

	functionality		and have at least one Safety-Observation record			with at least one record	
				2	Select a Safety-Observation record and click the "Delete" action	A confirmation dialog should appear if the record is not in workflow.	A confirmation dialog appears upon selecting a record and clicking "Delete".
				3	Confirm the delete action	The selected Safety-Observation record should be deleted, and the data grid should refresh without the deleted record	The selected Safety-Observation record is deleted, and the data grid refreshes without the deleted record.
TC_004	Verify search functionality	Safety – Observation - ListPage	User should be logged in with proper permissions and have multiple Safety-Observation records	1	Navigate to the Safety-Observation listing page	Safety-Observation listing page should load successfully with multiple records	Safety-Observation listing page loads successfully with all required filters and data grid
				2	Enter a search term in the search input field	Data grid should display only the records matching the search term	Data grid displays only the records matching the search term entered in the search input field.
				3	Clear the search input field	Data grid should display all records again	Safety-Observation listing page loads successfully with all required filters and data grid
TC_005	Verify export functionality	Safety – Observation - ListPage	User should be logged in with proper permissions and have Safety-Observation records.	1	Navigate to the Safety-Observation listing page	Safety-Observation listing page should load successfully with records	Safety-Observation listing page loads successfully with all required filters and data grid
				2	Click on the "PDF" or "Excel" export button	The application should generate and download the respective file format with the Safety-Observation data	The application generates and downloads the respective file format with the Safety-Observation data

6.2.3.2. Safety-Observation Form Page

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_01	Verify the page is opened in add mode and fields are available.	Safety – Observation - FormPage	User should have viewing and adding permission	**1**	Click on add button on Safety-Observation list page.	The Safety-Observation form should be opened with all the fields according to validations.	The Safety-Observation form opens with all the fields according to validations.
				2	Fill the details in form as required validations.	Only selected number of fields should be available based on user's interaction.	Only selected number of fields are available based on user's interaction.
TC_02	Verify the page is opened in edit mode and only editable fields are available.	Safety – Observation - FormPage	User should have viewing and editing permission	**1**	Click on edit button in Safety-Observation list page for that record.	The form page should be loaded with all the fields filled with the data.	The form page is loaded with all the fields filled with the data.
				2	Update the details in form.	Only the editable fields should be editable.	Only the editable fields are editable.
TC_03	Verify the submission of the form when the data is not filled or invalid.	Safety – Observation - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The alerts and messages should be displayed wherever the fields are bot filled or are invalid.	The alerts and messages are displayed wherever the fields are bot filled or are invalid.
TC_04	Verify the submission of the form when the data is fully filled and valid.	Safety – Observation - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The form should be submitted and the record of it should be displayed on Safety-Observation list page.	The form is submitted and the record of is displayed on Safety-Observation list page.
TC_05	Verify the edit page opened by user when it is in workflow.	Safety – Observation - FormPage	User should have viewing permission	**1**	Click on edit button in Safety-Observation list page for that record.	The form page should be loaded with all the fields filled with the data	The form page is loaded with all the fields filled with the data.
				2	Scroll down to see the action button visibility.	The submission buttons should not be visible so that user cannot edit data.	The submission buttons are not visible so that user cannot edit data.
TC_06	Verify the edit page opened by	Safety – Observation -	User should have viewing permission	**1**	Click on edit button in Safety-	The form page should be loaded with all the fields	The form page is loaded with all the fields filled with

	approver when it is in workflow.	FormPage			Observation list page for that record.	filled with the data	the data.
				2	Click on approve or disapprove button.	Only the approval/dis approval button should be visible and on clicking the record must be approved or disapproved.	Only the approval/dis approval button is visible and on clicking the record is approved or disapproved.

6.2.3.3. Safety-ObservationBackend

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Test Steps	Expected Result	Actual Result
TC_01	Verify GetObservationForList API	Safety – Observation – List Page	User should be logged in with proper permissions	Call the API GET /api/Observation/GetObservationForList with valid data	Safety-Observation listing should be successfully retrieved with all required filters and data grid	Safety-Observation listing was successfully retrieved with all required filters and data grid as expected.
TC_02	Verify GetObservationById API	Safety – Observation-FormPage	User should be logged in with proper permissions	Call the API GET /api/Observation/GetObservationById with valid data	Observation details should be successfully retrieved by ID	Observation details were successfully retrieved by ID as expected.
TC_03	Verify InsertObservation API	Safety – Observation FormPage	User should be logged in with proper permissions	Call the API POST /api/Observation/InsertObservation with valid data	Observation should be successfully inserted	Observation was successfully inserted as expected.
TC_04	Verify UpdateObservation API	Safety – Observation-FormPage	User should be logged in with proper permissions	Call the API PUT /api/Observation/UpdateObservation with valid data	Observation should be successfully updated	Observation was successfully updated as expected.
TC_05	Verify DeleteSafetyTransaction API	Safety - Transaction	User should be logged in with proper permissions	Call the API PUT /api/DeleteSafetyTransaction with valid data	Safety transaction should be successfully deleted based on the provided data	Safety transaction was successfully deleted based on the provided data as expected.
TC_06	Verify GetObservationStatus API	Safety – Safety Observation	-	Call the API GET /api/Observation/GetObservationStatus	Observation Status list should be successfully retrieved	Observation Status list was successfully retrieved as expected.
TC_07	Verify GetObservationNcrStatus API	Safety – Safety Observation	-	Call the API GET /api/Observation/GetObservationNcrStatus	Ncr Status List should be successfully retrieved	Ncr Status List was successfully retrieved as expected.
TC_08	Verify GetChecklistByProjid API	Safety – Safety Observation	-	Call the API GET /api/Observation/GetChecklistByProjid	List of checklists of the same project should be successfully retrieved	List of checklists of the same project was successfully retrieved as expected.
TC_09	Verify InsertObservationImage API	Safety – Safety Observation	User should be logged in with proper permissions	Call the API POST /api/Observation/InsertObservationImage	Observation image should be successfully inserted	Observation image was successfully inserted
TC_10	Verify InsertChecklistObservation API	Safety – Safety Observation	User should be logged in with proper permissions	Call the API POST /api/Observation/InsertChecklistObservation	Checklist Observation image should be successfully inserted	Checklist Observation image was successfully inserted

TC_11	Verify UpdateMRApproveDisapprove API	Safety – Observation-Formpage	User should be logged in with proper permissions	Call the API POST /api/Observation/UpdateMRApproveDisapprove with valid data	Observation should be successfully approved/disapproved	Observation was successfully approved/disapproved as expected.
TC_12	Verify GetSafetyApprovalAuditList API	Safety - Approval	User should be logged in with proper permissions	Call the API GET /api/Observation/GetSafetyApprovalAuditList with valid data	Approval audit list should be successfully retrieved with specified program ID and transaction ID	Approval audit list was successfully retrieved with specified program ID and transaction ID as expected.

6.2.4. NCR

6.2.4.1. Safety-NCR List Page

Test Case Id	Test Title/Test Case Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_001	Verify Ncr listing page	Safety – Ncr - ListPage	User should be logged in with proper permissions	1	Launch the application and navigate to the Ncr listing page	Safety-Ncr listing page should load successfully with all the required filters and data grid	Safety-Ncr listing page loads successfully with all required filters and data grid.
				2	Apply different filters (from date, to date, approval status, document status, status)	Data grid should display filtered data correctly	Safety-Ncr listing page loads successfully with all required filters and data grid.
				3	Click on the "Add" button	The user should be redirected to the Safety-Ncr add/edit page	The user is redirected to the Safety-Ncr add/edit page
TC_002	Verify Safety-Ncr add/edit functionality	Safety – Ncr - ListPage	User should be logged in with proper permissions	1	Navigate to the Safety-Ncr add/edit page	Safety-Ncr add/edit form should load successfully	Safety-Ncr add/edit form loads successfully.
				2	Fill in the required fields and submit the form	Safety-Ncr should be created/updated successfully, and the user should be redirected back to the listing page	Safety-Ncr is created/updated successfully upon submitting the form.
				3	Verify the newly created/updated Safety-Ncr in the listing page	The new/updated Safety-Ncr should be visible in the listing page	The new/updated Safety-Ncr is visible in the listing page.
TC_003	Verify Safety-Ncr delete functionality	Safety – Ncr - ListPage	User should be logged in with proper permissions and have at least one Safety-Ncr	1	Navigate to the Safety-Ncr listing page	Safety-Ncr listing page should load successfully with at least one record	Safety-Ncr listing page loads successfully with at least one record.
				2	Select a Safety-Ncr	A confirmation dialog should	A confirmation dialog appears upon

			record		record and click the "Delete" action	appear if the record is not in workflow.	selecting a record and clicking "Delete".
				3	Confirm the delete action	The selected Safety-Ncr record should be deleted, and the data grid should refresh without the deleted record	The selected Safety-Ncr record is deleted, and the data grid refreshes without the deleted record.
TC_004	Verify search functionality	Safety – Ncr - ListPage	User should be logged in with proper permissions and have multiple Safety-Ncr records	1	Navigate to the Safety-Ncr listing page	Safety-Ncr listing page should load successfully with multiple records	Safety-Ncr listing page loads successfully with all required filters and data grid
				2	Enter a search term in the search input field	Data grid should display only the records matching the search term	Data grid displays only the records matching the search term entered in the search input field.
				3	Clear the search input field	Data grid should display all records again	Safety-Ncr listing page loads successfully with all required filters and data grid
TC_005	Verify export functionality	Safety – Ncr - ListPage	User should be logged in with proper permissions and have Safety-Ncr records.	1	Navigate to the Safety-Ncr listing page	Safety-Ncr listing page should load successfully with records	Safety-Ncr listing page loads successfully with all required filters and data grid
				2	Click on the "PDF" or "Excel" export button	The application should generate and download the respective file format with the Safety-Ncr data	The application generates and downloads the respective file format with the Safety-Ncr data

6.2.4.2. Safety-NCR Form Page

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Sr. No.	Test Steps	Expected Result	Actual Result
TC_01	Verify the page is opened in add mode and fields are available.	Safety – Ncr - FormPage	User should have viewing and adding permission	**1**	Click on add button on Safety-Ncr list page.	The Safety-Ncr form should be opened with all the fields according to validations.	The Safety-Ncr form opens with all the fields according to validations.
				2	Fill the details in form as required validations.	Only selected number of fields should be available based on user's interaction.	Only selected number of fields are available based on user's interaction.
TC_02	Verify the page is opened in edit mode and only editable fields are available.	Safety – Ncr - FormPage	User should have viewing and editing permission	**1**	Click on edit button in Safety-Ncr list page for that record.	The form page should be loaded with all the fields filled with the data.	The form page is loaded with all the fields filled with the data.
				2	Update the details in form.	Only the editable fields should be editable.	Only the editable fields are editable.
TC_03	Verify the submission of the form when the data is not filled or invalid.	Safety – Ncr - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The alerts and messages should be displayed wherever the fields are bot filled or are invalid.	The alerts and messages are displayed wherever the fields are bot filled or are invalid.
TC_04	Verify the submission of the form when the data is fully filled and valid.	Safety – Ncr - FormPage	User should have viewing, adding and editing permission	**1**	Click on submit button in the form page.	The form should be submitted and the record of it should be displayed on Safety-Ncr list page.	The form is submitted and the record of is displayed on Safety-Ncr list page.
TC_05	Verify the edit page opened by user when it is in workflow.	Safety – Ncr - FormPage	User should have viewing permission	**1**	Click on edit button in Safety-Ncr list page for that record.	The form page should be loaded with all the fields filled with the data	The form page is loaded with all the fields filled with the data.
				2	Scroll down to see the action button visibility.	The submission buttons should not be visible so that user cannot edit data.	The submission buttons are not visible so that user cannot edit data.
TC_06	Verify the edit page opened by approver when it is in workflow.	Safety – Ncr - FormPage	User should have viewing permission	**1**	Click on edit button in Safety-Ncr list page for that record.	The form page should be loaded with all the fields filled with the data	The form page is loaded with all the fields filled with the data.
				2	Click on	Only the	Only the

					approve or disapprove button.	approval/dis approval button should be visible and on clicking the record must be approved or disapproved.	approval/dis approval button is visible and on clicking the record is approved or disapproved.

6.2.4.3. Safety-NCR Backend

Test Case Id	Test Title/Test Cases Description	Component	Prerequisite	Test Steps	Expected Result	Actual Result
TC_01	Verify GetNcrForList API	Safety – Ncr – List Page	User should be logged in with proper permissions	Call the API GET /api/Ncr/ GetNcrForList with valid data	Safety-Ncr listing should be successfully retrieved with all required filters and data grid	Safety-Ncr listing was successfully retrieved with all required filters and data grid as expected.
TC_02	Verify GetNcrById API	Safety – Ncr-FormPage	User should be logged in with proper permissions	Call the API GET /api/Ncr/ GetNcrById with valid data	Ncr details should be successfully retrieved by ID	Ncr details were successfully retrieved by ID as expected.
TC_03	Verify InsertNcr API	Safety – Ncr FormPage	User should be logged in with proper permissions	Call the API POST /api/Ncr/ InsertNcr with valid data	Ncr should be successfully inserted	Ncr was successfully inserted as expected.
TC_04	Verify UpdateNcr API	Safety – Ncr-FormPage	User should be logged in with proper permissions	Call the API PUT /api/Ncr/UpdateNcr with valid data	Ncr should be successfully updated	Ncr was successfully updated as expected.
TC_05	Verify DeleteSafetyTransaction API	Safety - Transaction	User should be logged in with proper permissions	Call the API PUT /api/DeleteSafety Transaction with valid data	Safety transaction should be successfully deleted based on the provided data	Safety transaction was successfully deleted based on the provided data as expected.
TC_06	Verify GetSafetyNcrType API	Safety – Safety Ncr	-	Call the API GET /api/Ncr/GetSafetyNcrType	Ncr Type list should be successfully retrieved	Ncr Type list was successfully retrieved as expected.
TC_07	Verify GetSafetyNcrStatus API	Safety – Safety Ncr	-	Call the API GET /api/Ncr/GetSafetyNcrStatus	Ncr Status List should be successfully retrieved	Ncr Status List was successfully retrieved as expected.
TC_08	Verify GetObsWithSafetyNcr API	Safety – Safety Ncr	-	Call the API GET /api/Ncr/GetObsWithSafetyNcr	List of observations closed as NCR should be successfully retrieved	List of observations closed as NCR was successfully retrieved.
TC_09	Verify GetUserByProjIdForN	Safety – Safety Ncr	-	Call the API GET /api/Ncr/GetUserByProjIdForNcr	List of users of that project should be successfully	List of users of that project was successfully retrieved.

	cr API				retrieved	
TC_10	Verify UpdateMRApproveDisapprove API	Safety – Ncr-Formpage	User should be logged in with proper permissions	Call the API POST /api/Ncr/UpdateMRApproveDisapprove with valid data	Ncr should be successfully approved/disapproved	Ncr was successfully approved/disapproved as expected.
TC_11	Verify GetSafetyApprovalAuditList API	Safety - Approval	User should be logged in with proper permissions	Call the API GET /api/Ncr/GetSafetyApprovalAuditList with valid data	Approval audit list should be successfully retrieved with specified program ID and transaction ID	Approval audit list was successfully retrieved with specified program ID and transaction ID as expected.

Chapter 7: Future Enhancement

- **Integration with Business Intelligence (BI) Tools:**
  - Integrate the ERP system with advanced BI tools such as Tableau, Power BI, or QlikSense to enable in-depth data analysis, visualization, and reporting capabilities.
  - Allow users to create custom dashboards, drill-down reports, and interactive data visualizations to gain actionable insights into business performance and trends.

- **Advanced Data Analytics Features:**
  - Implement machine learning algorithms and predictive analytics capabilities within the ERP system to analyze historical data, forecast future trends, and optimize decision-making processes.

- **Development of Mobile Application:**
  - Develop a dedicated mobile application for the ERP system to provide users with on-the-go access to key functionalities and data.
  - Enable features such as mobile approvals, real-time notifications, and remote data entry to enhance productivity and flexibility for users working in the field or on-site.

- **User Feedback and Iterative Development:**
  - Establish a feedback loop with users to gather input, suggestions, and feature requests for continuous improvement of the ERP system.
  - Prioritize enhancements based on user feedback, business priorities, and emerging market trends to ensure the ERP system remains aligned with the evolving needs of Riviera Infraprojects.

References

- https://www.npmjs.com/
- https://mui.com/material-ui/
- https://react.dev/
- https://www.postgresql.org/docs/
- https://learn.microsoft.com/en-us/aspnet/core/data/entity-framework-6?view=aspnetcore-8.0
- https://stackoverflow.com/

adarsh1

ORIGINALITY REPORT

6%	5%	0%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Institute of Technology, Nirma University Student Paper	4%
2	Submitted to Charotar University of Science And Technology Student Paper	1%
3	Submitted to Sardar Vallabhbhai National Inst. of Tech.Surat Student Paper	<1%
4	Submitted to Kingston University Student Paper	<1%
5	Submitted to University of Wales Institute, Cardiff Student Paper	<1%
6	docshare.tips Internet Source	<1%
7	fastercapital.com Internet Source	<1%
8	technodocbox.com Internet Source	<1%

9	pdfcoffee.com Internet Source	<1%
10	Submitted to Fiji National University Student Paper	<1%
11	Submitted to University of New York in Tirana Student Paper	<1%
12	Submitted to University of Technology, Sydney Student Paper	<1%
13	blogs.sap.com Internet Source	<1%
14	Submitted to deped-pasig Student Paper	<1%
15	ethesis.nitrkl.ac.in Internet Source	<1%
16	Submitted to West Herts College Student Paper	<1%
17	www.coursehero.com Internet Source	<1%
18	www.cybersecurity.my Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off